CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

FACULTY OF APPLIED SCIENCES

DEPARTMENT OF MATHEMATICAL SCIENCES

SEMESTER 3 B.Tech. CE, IT, CSE

DISCRETE MATHEMATICS AND ALGEBRA

MA253

UNIT : 3

Graph Theory

INTRODUCTION TO GRAPH THEORY

In a hockey tournament there are 8 teams which are denoted by S,T,U,V,W,X,Y,Z. Let the following games has been played.

S → X, Z	T → W, X, Z
U → Y, Z	V → W,Y
W →T, V, Y	X → S, T
Y → U, V, W	Z → S, T, U

We may illustrate this situation by following diagram:

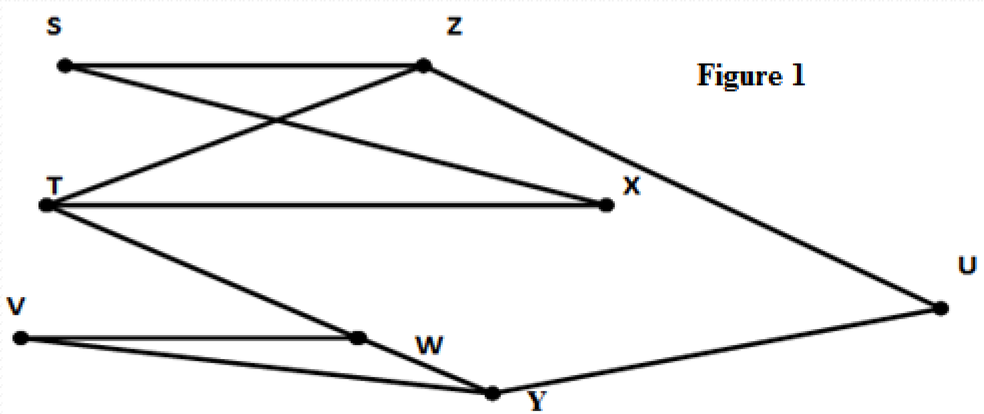


Figure 1

Or one can draw in the following way:

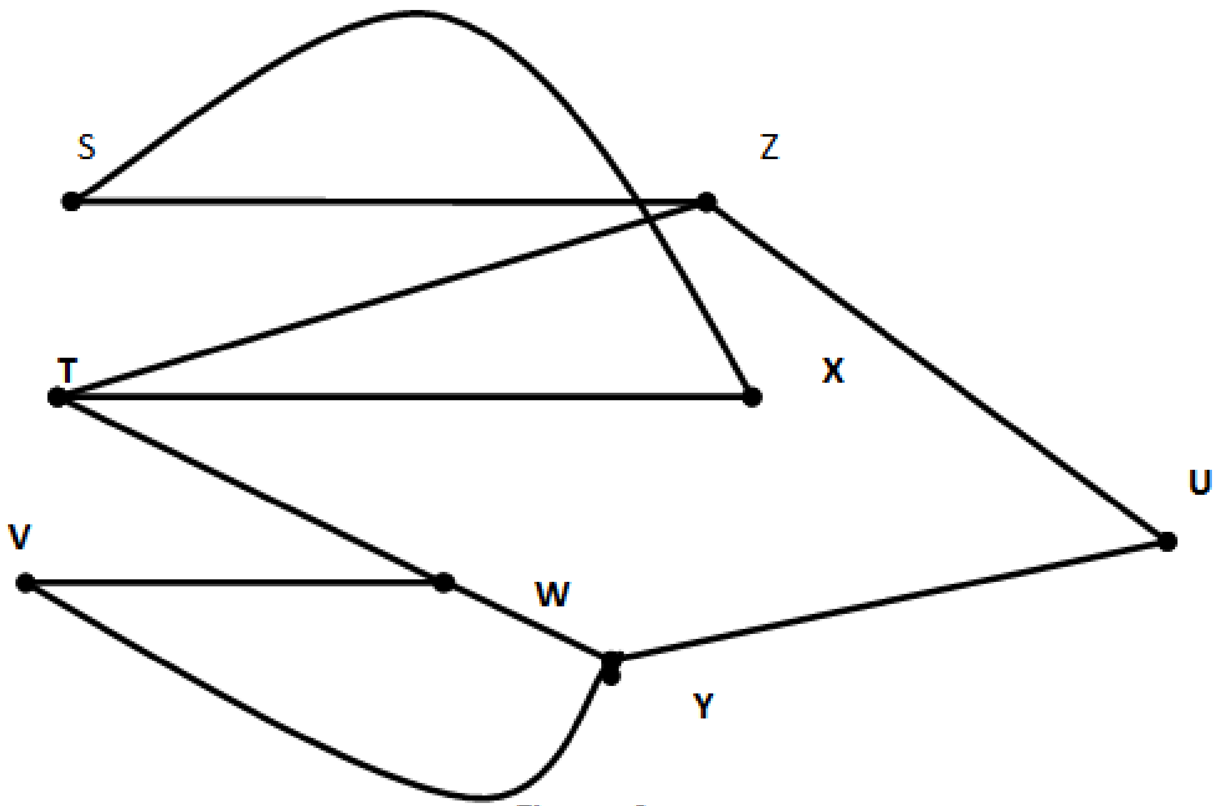


Figure 2

Graphical Representation

The team are represented by dots. Two such dots are join by line segment wherever the corresponding teams have played each other. In the figure-1, dots have been joined by straight line while in the figure-2, some of the line segment are straight but some are not. But since we are simply interested in which games have been played. The manner in which the pair of dots is joined is of no importance so it does not matter whether the line segments are straight or not. Many real world situations can be described by means of such type of diagram which are called **graph.**

Definition of GRAPH:

A graph G = (V(G), E(G)) consist of two finite set V(G) and E(G), where V(G) = The vertex set of graph, often denoted by just V which is non-empty set of elements **called vertices or nodes** and E(G) = The edge set of the graph often denoted by E which is possibly empty set elements called **edges or arc** such that **e** in E assigned an unordered pair of vertices (u, v) called the **end vertices of e.**

Order of a Graph G:

The order of a graph G, written as n(G), is no. of vertices in the graph G. It is denoted by |V(G)|.

The size of the graph G:

The size of the graph G, written as e(G), is no. of edges in graph. It is denoted by |E(G)|.

For the above example,

V={S, T, U, V, W, X, Y, Z}

E={e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9, e_{10}}

$e_1 \leftrightarrow$ (S,Z)

$e_2 \leftrightarrow$ (S,X) **(UNORDERED PAIR)**

$e_3 \leftrightarrow$ (T,Z)

$e_4 \leftrightarrow$ (T,X)

$e_5 \leftrightarrow$ (T,W)

$e_6 \leftrightarrow$ (U,Y)

$e_7 \leftrightarrow$ (Y,V)

$e_8 \leftrightarrow$ (V,W)

$e_9 \leftrightarrow$ (W,Y)

$e_{10} \leftrightarrow$ (U,Z)

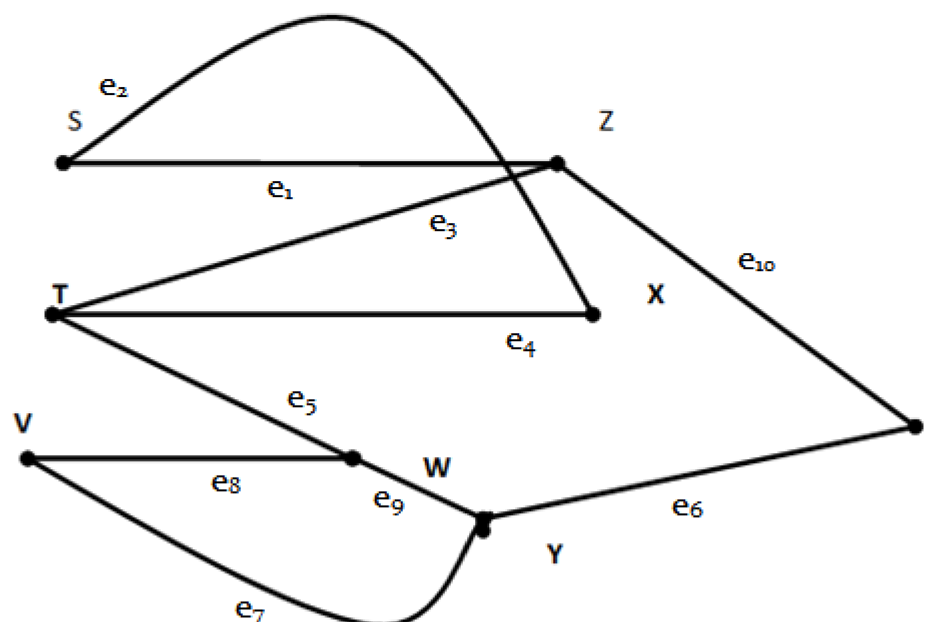


Figure 2

The order of the above Graph is 8 and the size of Graph is 10.

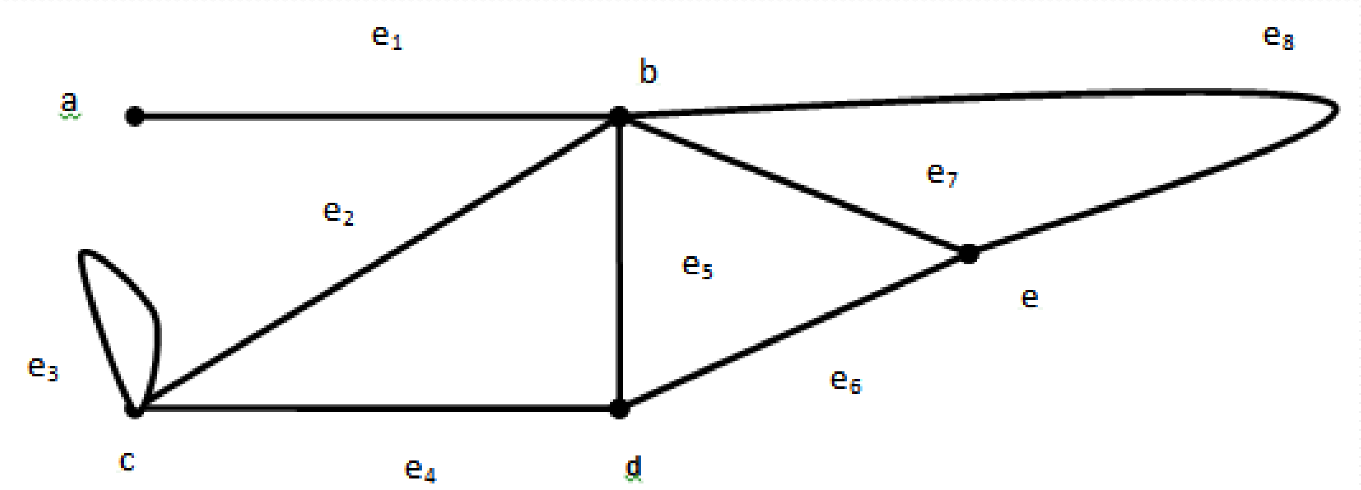
Remark: If **e** is an edge with end vertices u and v then e is said to join u and v.

Example-1: Let G=(V,E), where V={a, b, c, d, e} and E={e_1, e_2, e_3, e_4, e_5,…, e_8} and the ends of the edges are given by

$e_1 \leftrightarrow$ (a,b), $e_2 \leftrightarrow$ (b,c), $e_3 \leftrightarrow$ (c,c), $e_4 \leftrightarrow$ (c,d),

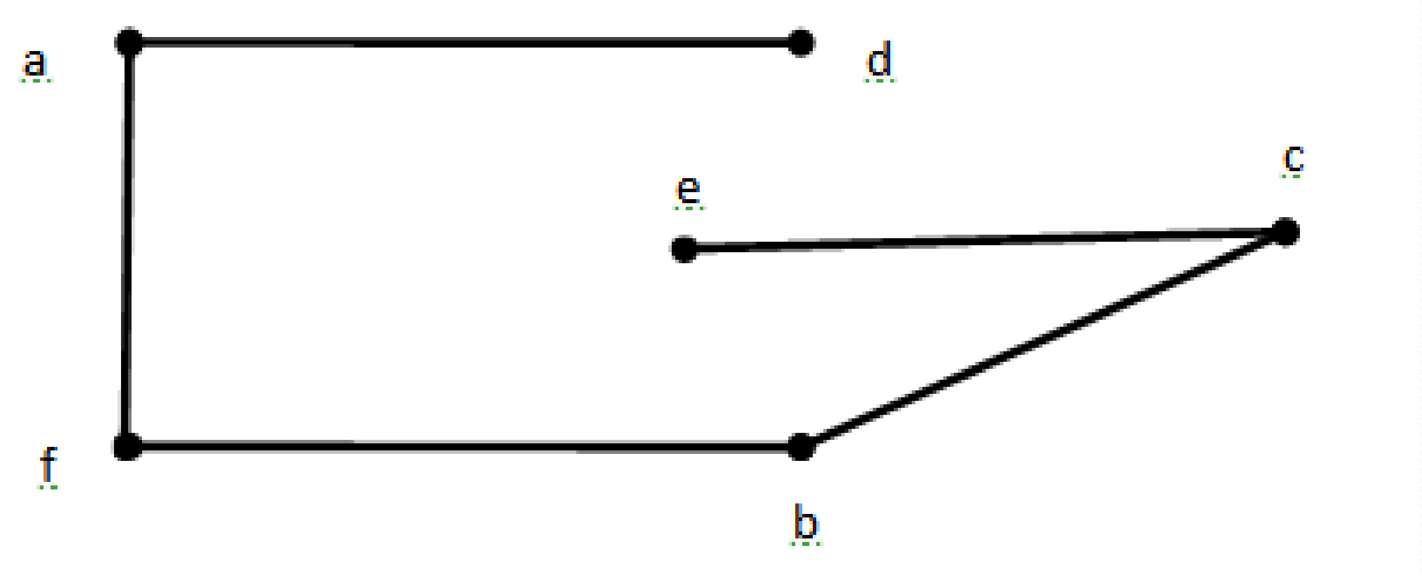
$e_5 \leftrightarrow$ (b,d), $e_6 \leftrightarrow$ (d,e), $e_7 \leftrightarrow$ (b,e), $e_8 \leftrightarrow$ (b,e). Draw a diagram.

Solution:



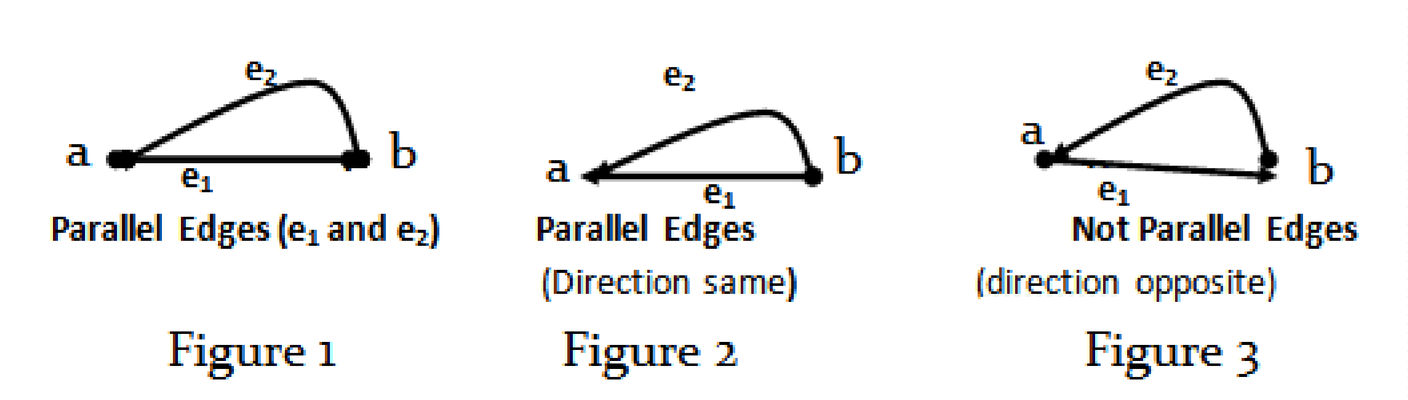
Example-2: Draw a diagram for each of the following graph G(V,E), where V={a, b, c, d, e ,f} and E={(a, d), (a, f) ,(b, c), (b ,f),(c, e)}.

Solution:



Loop (Self loop): An edge of graph which join a vertex to itself is said to be a loop i.e., an edge of the graph having identical end vertices is said to be loop. In figure given below **e** is the loop:

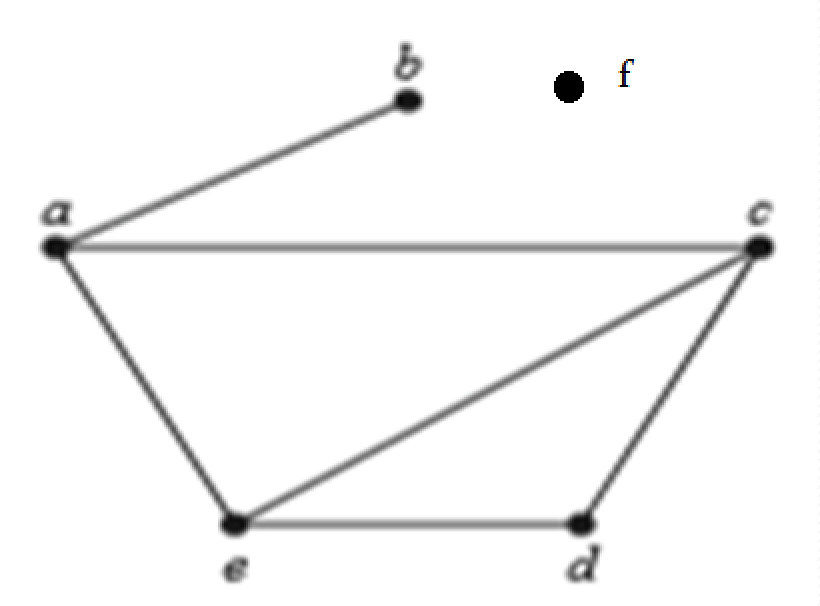
Parallel Edge or Multi Edge: Let G be a graph, if two (or more) edges of G have same vertices then edges are said to be parallel.



Isolated Vertex: A vertex of a graph which is not end of any edge is said to be isolated vertex.

Adjacent Vertices: Two vertices of a graph which are joined by an edge are said to be adjacent vertices or neighbours to each other.

Vertex	Adjacent Vertex
a	b, c, e
b	a
c	a, e, d
d	c ,e
e	a, d, c
f (Isolated vertex)	----------



Neighbour of a vertex: Let v be a vertex of a graph G. If vertex v is joined by an edge to a vertex u of g, then u is said to be neighbour of vertex v**. In graph G neighbour of a vertex a are b, c, e.**

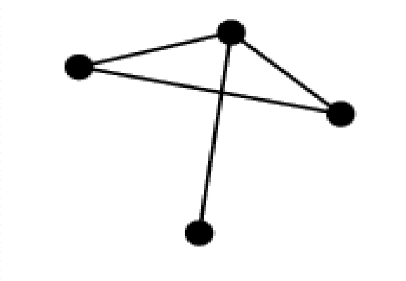
Neighbourhood set of vertex: The set of all neighbour for fixed vertex v of a graph G is said to be neighbourhood set for vertex v. It is denoted by N(v). **In graph G, N(a)={b ,c, e}**

Incident Edge: An edge e of a graph G is said to be incident with vertex v if v is an end vertex of e, i.e., edge connected to a vertex v.

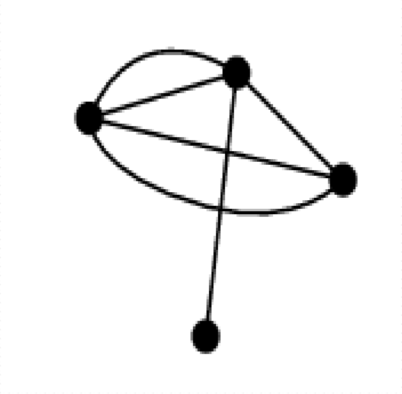
Adjacent Edges: Two edges e and f of the graph G which are incident with a common vertex is said to be adjacent edges.

<u>TYPES OF GRAPH</u>

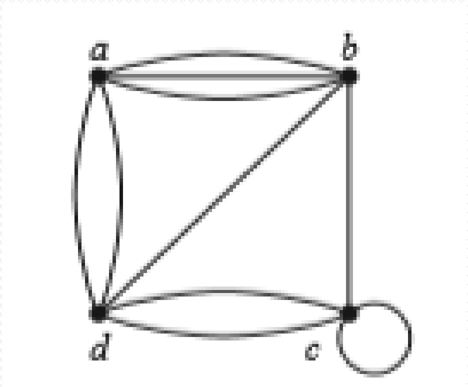
<u>SIMPLE GRAPH</u>: A Graph which neither contains loop nor parallel edges (end points are same), is known as **Simple Graph**. For example, the following graph is a simple graph.



<u>MULTI GRAPH</u>: A Graph which contains multi edges(parallel) is known **Multi Graph**.



<u>PSEUDOGRAPH</u>: A graph in which loops and multi edges are allowed is called a **Pseudo graph**.



<u>NULL GRAPH</u>: Let G = (V, E) be a graph, then it is known as **Null Graph** if i.e. there is no edges in graph or all the vertices of a graph are isolated.

<u>TRIVIAL GRAPH</u>: A graph having one vertex and no edge is known as **trivial graph.**

Remark: Every Trivial graph is a null graph but every null graph need not to be trivial graph.

FINITE GRAPH: Let G be a graph, then it is said to be **finite,** if it has finite number of vertices and finite number of edges.

Clearly, a graph with finite number of vertices must contain finite number of edges; this implies a graph with finite number of vertices will be automatically finite.

A graph which is not finite is known as **infinite graph.**

DIRECTED GRAPH (Digraph): A **directed graph** (or **digraph**) is a graph that is made up of a set of vertices connected by edges, where the edges have a direction associated with them.

The notation e→ (u, v) says e is an directed edge with initial vertex u and terminal or end vertex v.

<u>MIXED GRAPH</u>: A Graph is known as Mixed Graph if some edges are directed and some edges are undirected as shown in the figure above.

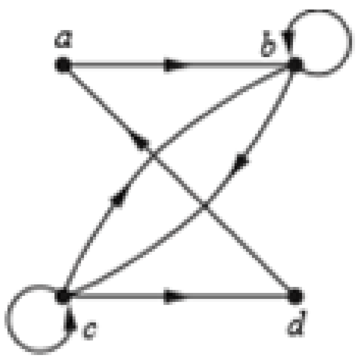


Figure: Directed Graph

(As all the edges are having direction)

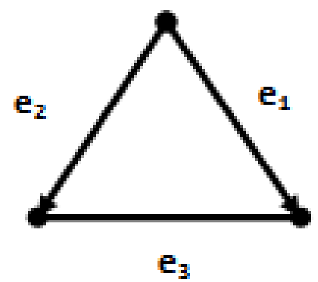


Figure: Mix Graph (e_1 and e_2 are directed edges

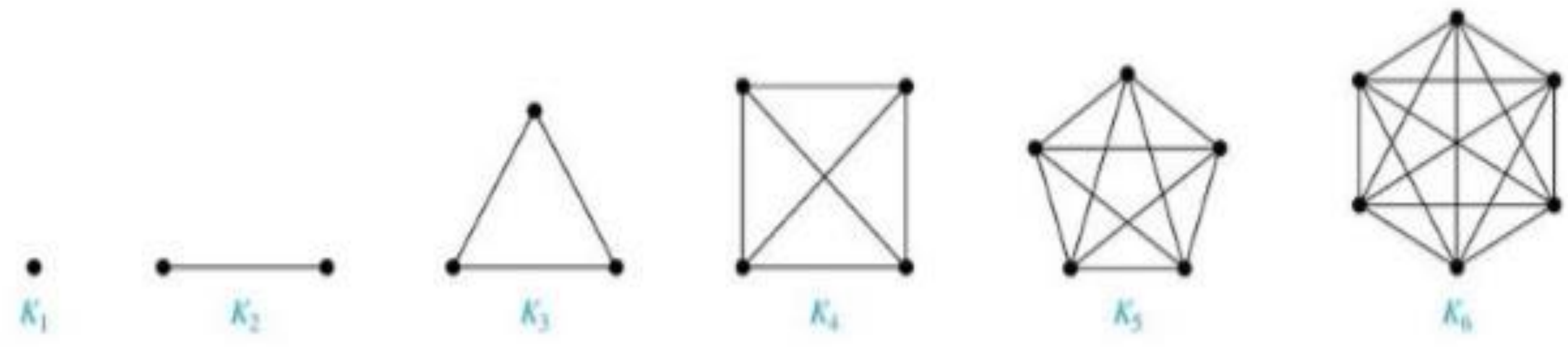
but e_3 is not a directed edge)

Graph Terminology			
Type	*Edges*	*Multiple Edges Allowed?*	*Loops Allowed?*
Simple graph	Undirected	No	No
Multigraph	Undirected	Yes	No
Pseudograph	Undirected	Yes	Yes
Simple directed graph	Directed	No	No
Directed multigraph	Directed	Yes	Yes
Mixed graph	Directed and undirected	Yes	Yes

WEIGHTED GRAPH: A weighted graph is a graph G in which each edge e has been assign a real number w(e). **w(e) is said to be weight(or length) of edge.**

COMPLETE GRAPH: A simple graph in which each pair of distinct vertices is joined by an edge is said to be **complete graph**. A complete graph with n vertices is denoted by $\mathbf{K_n}$.

Complete graph with one, two, three, four, five, six vertices.



Remark: The number of edges of a complete graph K_n is $\frac{n(n-1)}{2}$.

Example: Find the number of edges in the graphs K_{12} and K_{15}.

Solution: The number of edges of a K_n is $\frac{n(n-1)}{2}$.

For K_{12}, Here n=12, No. of edges in K_{12} is $\frac{(12)(12-1)}{2}$**=66.**

For K_{15} , Here n=15, No. of edges in K_{15} is $\frac{(15)(15-1)}{2}$**=105.**

<u>BIPARTITE GRAPH:</u> Let G be a graph. If vertex set V of G can be divided into two non empty disjoint subset X and Y (i.e. V= XUY, $X \neq \phi, Y \neq \phi, X \cap Y = \phi$) such that each edge of G has one end point in X and one end point in Y then G is said to be bipartite graph.

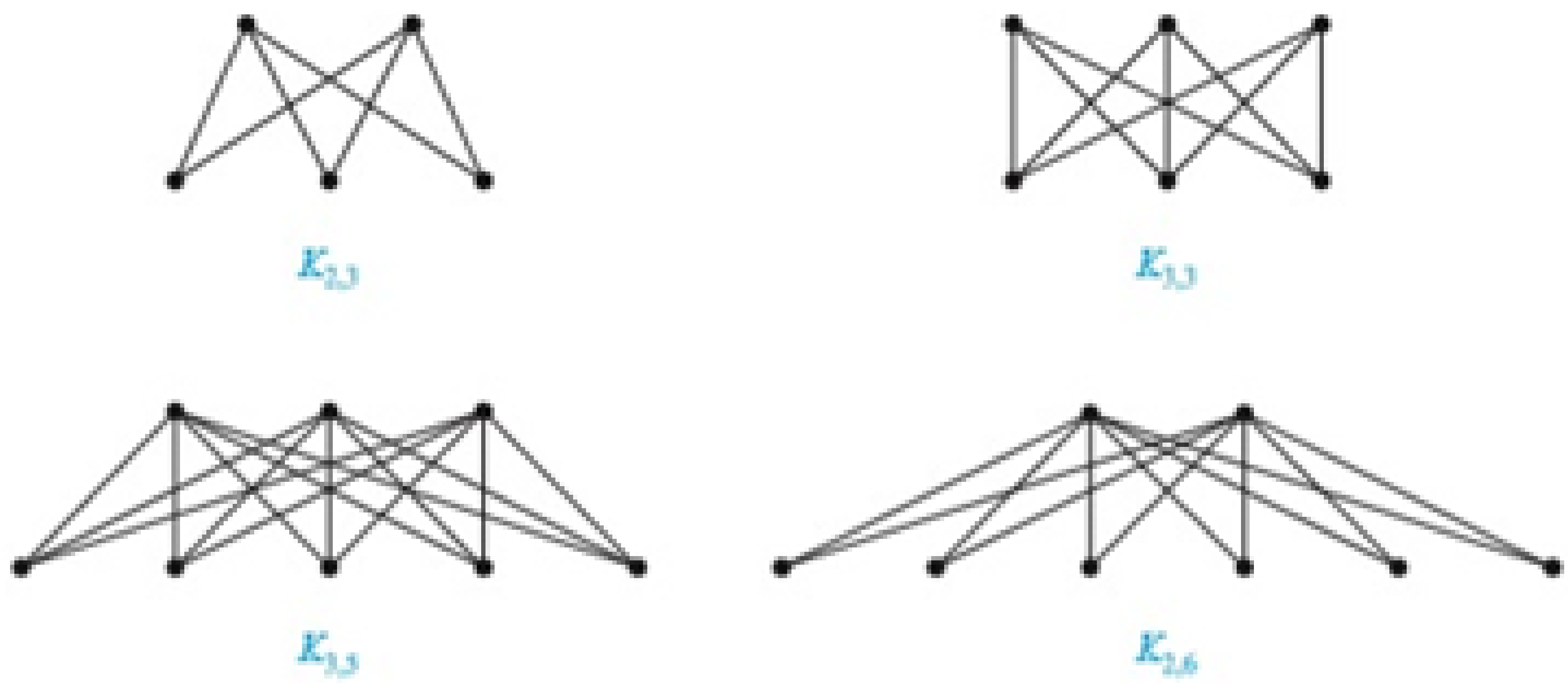
The bipartite graph is denoted by $K_{m,n}$ where m, n denotes the number of vertices in X and Y.

Note: Bipartite graph does not have any loop.

<u>COMPLETE BIPARTITE GRAPH</u>: A simple bipartite graph with partition V=XUY is said to be complete bipartite graph if every vertex in X is joined to each vertex of Y.

Note: The number total edges in a complete bipartite graph are mn because each m vertices is connected to each of n vertices. It is denoted by $K_{m,n}$, where $m \leq n$.

The complete bipartite graph $K_{2,3}$, $K_{3,3}$, $K_{3,4}$, $K_{2,6}$ are display in the below figure:



Complete Bipartite Graph $K_{2,3}$, $K_{3,3}$, $K_{3,4}$, $K_{2,6}$

FOR UNDIRECTED GRAPH

DEGREE OF VERTEX: Let G be a graph and v be any vertex of G. The number of edges of G connected with the vertex v is called degree of a vertex.

Loop is counted twice.

It is denoted by d(v) or deg(v) or $d_G(v)$.

Remark: The degree of a vertex of a **simple graph** G on n vertices cannot exceed n-1.

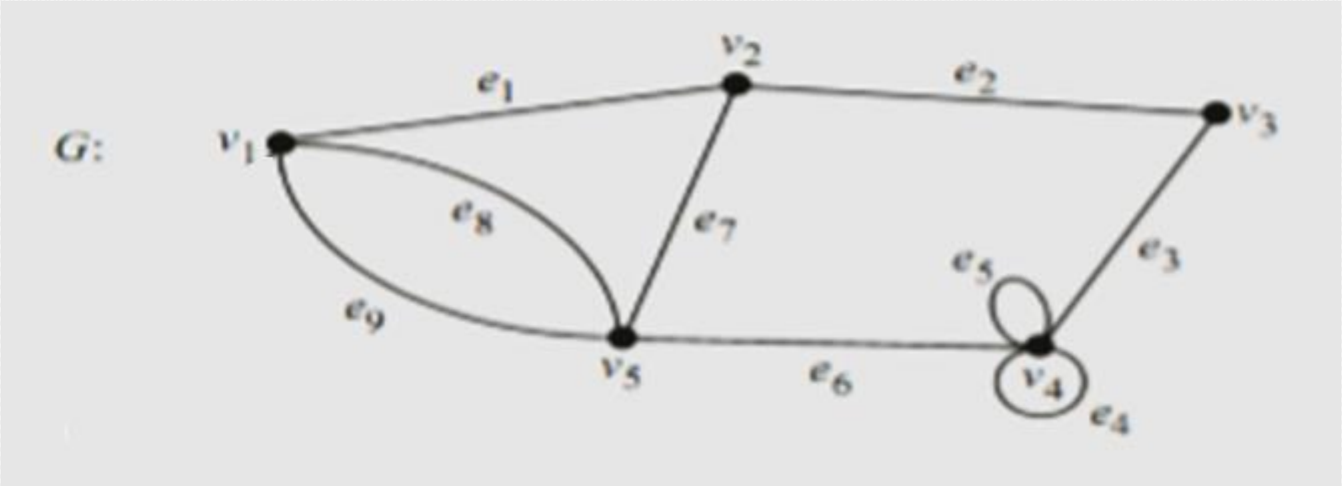
ODD VERTEX & EVEN VERTEX: In a graph a vertex with odd degree is said to be **odd vertex** and vertex with even degree is said to be **even vertex.**

Remark:

The degree of vertex is also known as its **valency.**

2. The degree of isolated vertex is zero.
3. The vertex with degree one is known as **Pendant vertex**.

Example: Find the degree of the vertices in the given graph. Also find the even and odd vertex in the given graph.



Solution:

Sr .no	Vertex	degree $d_i(v)$	Odd/Evevn Vertex
1	v_1	3	Odd
2	v_2	3	Odd
3	v_3	2	Even
4	v_4	6	Even
5	v_5	4	Even

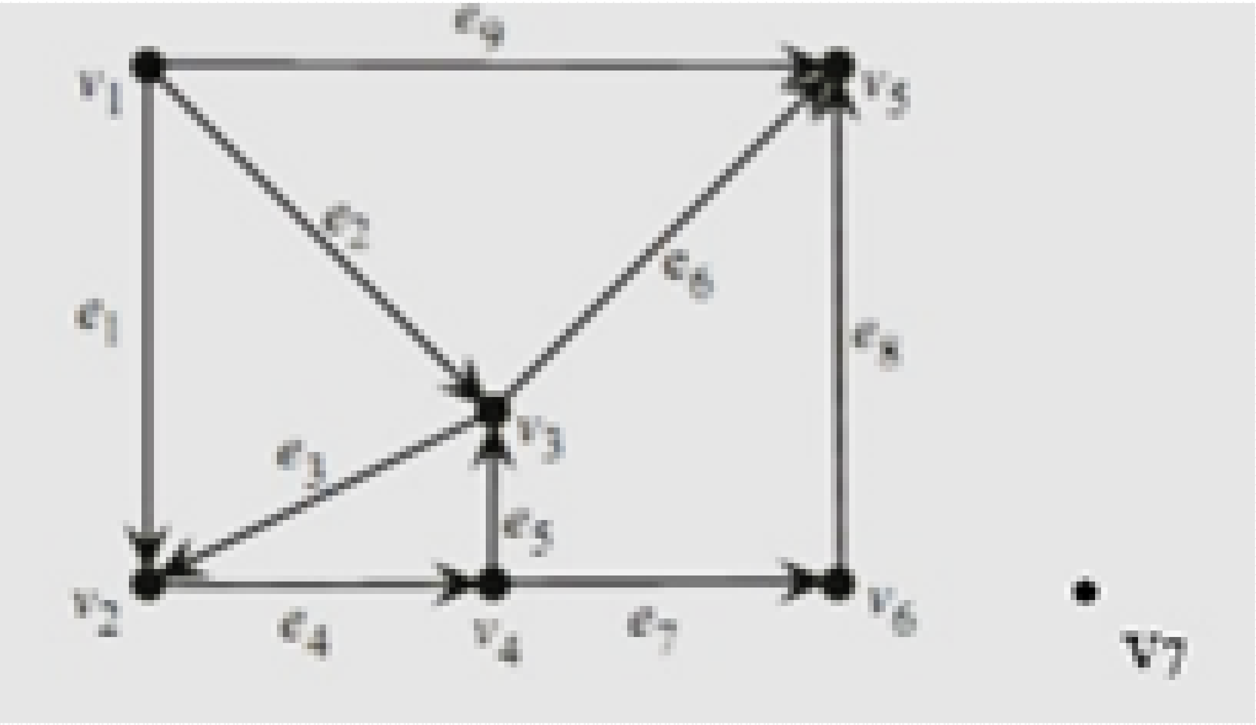
FOR DIRECTED GRAPH

INDEGREE OF VERTEX: The in degree of a vertex in directed graph is the number edges ending at it and it is denoted by indeg(v) or $d^+(v)$.

OUT DEGREE OF A VERTEX: The out degree of a vertex v in a directed graph is the number of edges beginning from it and it is denoted by outdeg(v) or $d^-(v)$.

The total degree is the sum of indegree and out degree.

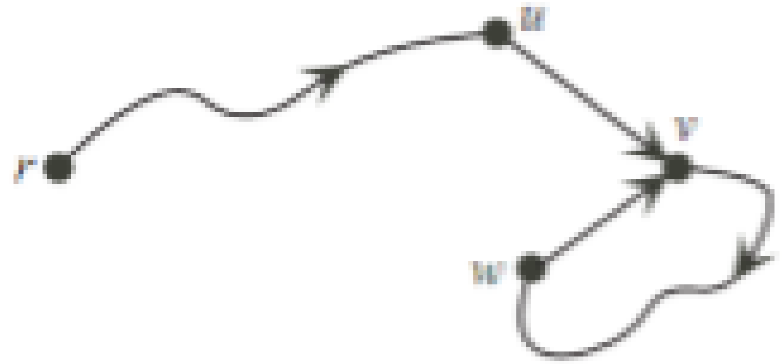
Example : Find the in degree and out degree of the vertices in following graph. Also find the even and odd vertex in the graph.



Solution:

Sr .no	Vertex	In degree $d_i(v)$	Out degree $d_o(v)$	Total degree $d_G(v)$	Odd/Evevn Vertex
1	v_1	0	3	3	Odd
2	v_2	2	1	3	Odd
3	v_3	2	2	4	Even
4	v_4	1	2	3	Odd
5	v_5	3	0	3	Odd
6	v_6	1	1	2	Even
7	v_7	0	0	0	Even

Example: Find the in degree and out degree of the vertices in the given graph.



Solution:

Sr .no	Vertex	In degree $d_i(v)$	Out degree $d_o(v)$	Total degree $d_G(v)$	Odd/Even Vertex
1	r	0	1	1	Odd
2	u	1	1	2	Even
3	v	2	1	3	Odd
4	w	1	1	2	Even

<u>THEOREM</u>: If G=(V,E) be a directed graph with e edges, then

$$\sum_{v \in V} \deg^{+}{}_{G}(v) = \sum_{v \in V} \deg^{-}{}_{G}(v) = e$$

i.e., the sum of the out degree of the vertices of a directed graph equals the sum of in degree of vertices which equals the number of edges in G.

<u>THEOREM</u>: (FIRST THEOREM OF GRAPH THEORY OR HANDSHAKING THEOREM):

If G=(V,E) be an undirected graph with e edges, then

$$\sum_{v \in V} \deg_{G}(v) = 2e.$$

i.e., The sum of degrees of the vertices in an undirected graph is 2e.

Corollary:
Undirected graph has an even number of vertices of odd degree.

Example: Is there a simple undirected graph corresponding to the following degree sequences (1, 1,2, 3)?

Solution:

In any graph, we know that there exists always even number of vertices of odd degree.

But here in given graph total number of vertices are 3 i.e., odd. Hence there exists no such graph corresponding to this given degree sequence.

Example: How many edges does an undirected graph have if it degree sequence is 4, 3, 3, 2, 2?

Solution:

Using the theorem of Graph theory, we know that

the sum of the degree of vertices of a graph = 2e no. of Edges

Here,

$4+3+3+2+2=2e$

So, $14=2e$

Therefore, $e=7$, number of edges

Thus, 7 edges required to construct a graph having as degree sequence is 4, 3, 3, 2, 2.

Example: Determine the number of edges in an undirected graph with 6 nodes(vertices), two of degree 4 and four of degree 2.

Solution:

The total degree of the vertex =2*4+4*2=16 .

We know that the sum of the degree of vertices in a graph = 2e number of edges.

So, Number of edges=16/2=8.

Thus, the number of edges in a graph with 6 nodes(vertices), two of degree 4 and four of degree 2 is 8.

Example: A non directed graph G has 8 edges. Find the number of vertices if the degree of each vertex is 2.

Solution:

Suppose that there are n vertices say $v_1, v_2, v_3, v_4, \dots, v_n$.

The degree of each vertices is same and it is 2,

i.e. $\deg(v_1) = \deg(v_2) = \dots = \deg(v_n) = 2$.

Now from Handshaking theorem, we have,

Sum of degree of vertices =2*No of edges.

$\deg(v_1) + \deg(v_2) + \dots + \deg(v_n) = 2*8$

$2+2+\dots+2(n \text{ times}) = 16$

$2n = 16 \Rightarrow n = 8$.

The number of required vertices is 8.

Example: Suppose G is non directed graph with 12 edges. If it has 6 vertices each of degree 3 and rest have Equal degree less than 3. Find the minimum and maximum number of vertices G can have.

Solution: We know that the

Sum of degree of all vertices = 2* No. of edges=2*12=24.

Total degree= 6*3 +$\sum \mathrm{d}(v)$=24,

$\therefore \sum \mathrm{d}(v) = 24 - 18 = 6.$

Now if degree of rest of the vertices(n)is 2 then 2n=6⇒ n = 3 and in total 9 vertices.

And if degree of rest of the vertices(n) is 1 then n = 6 and in total 12 vertices.

G can have 9 and 12 Minimum and maximum number of vertices respectively.

<u>REGULAR GRAPH</u>: In a graph if all vertices are of same degree then graph is said to be regular graph.

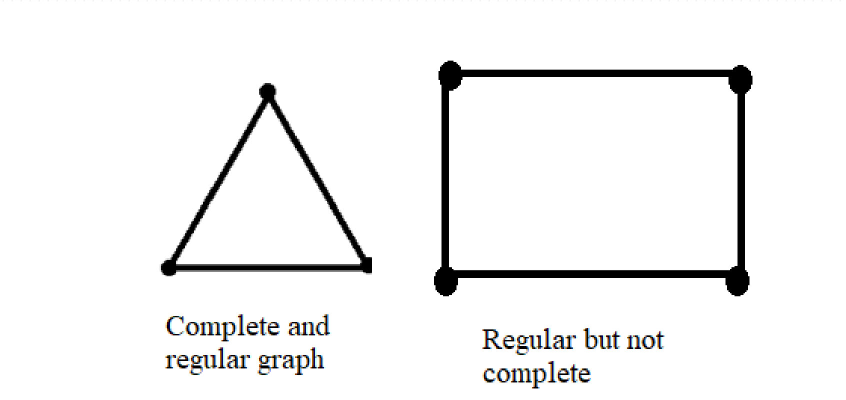
Remarks:

1)If every vertex has degree r, then graph is called regular graph of degree r.

2)Every null graph is a regular graph of degree zero.

3)If a graph has n vertex and is regular of degree r, then it has rn/2 edge.

4)All regular graphs need not to be complete but all complete graph are regular.



K-REGULAR GRAPH:

If The degree of every vertex of graph G is K, then G is said to be K-regular graph.

ISOMORPHIC GRAPH:

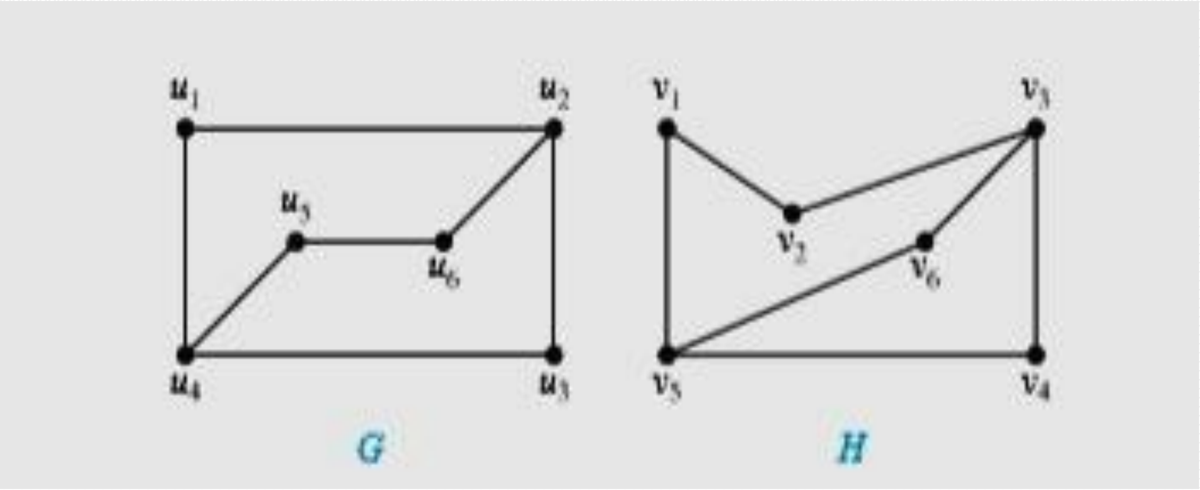
Let $G_1(V_{1,} E_1)$ and $G_2(V_{2,} E_2)$ be two graph. These two graphs are said to be isomorphic if there exist one-to-one correspondence between their vertices as well as edges.

Remarks:

1)By definition two isomorphic graphs must have Equal number of vertices with same degree.

2)One-to-one correspondence between vertex set as well edge set.

Example: Check whether the given graphs are isomorphic or not.

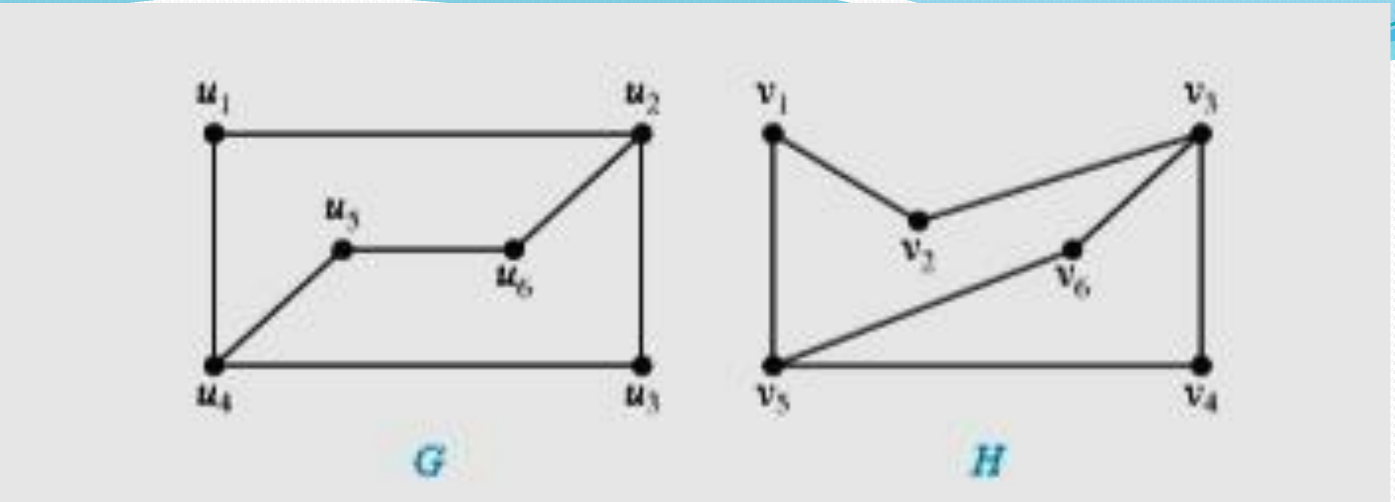


<u>Solution:</u>

CONDITION 1: Check no. of vertices and edges in both the graph are same or not. If same, then check condition 2.

CONDITION 2: Find one to one correspondence between vertex set and edge set of both graph.

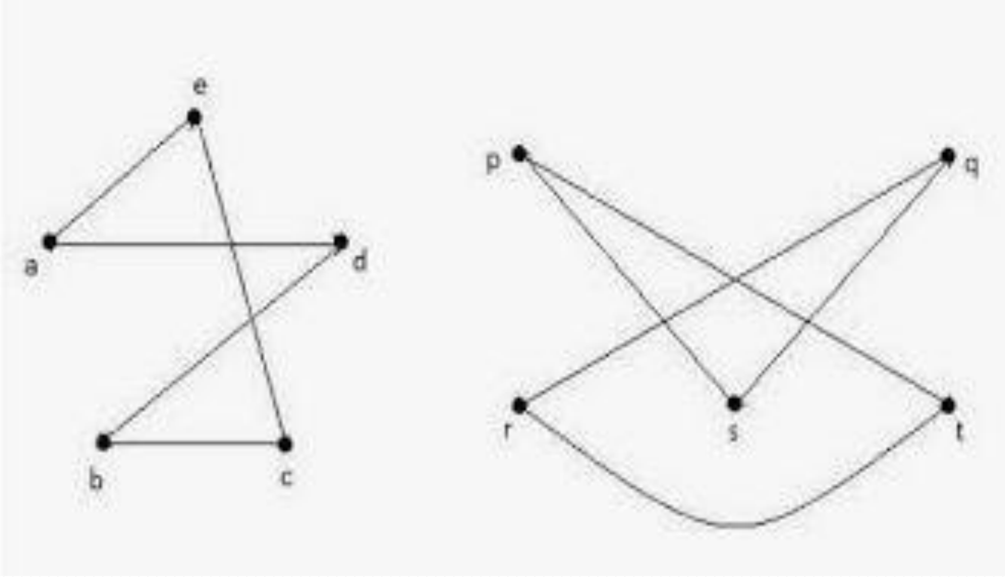
Here both G and H have six vertices and seven edges. Both have four vertices of degree two and two vertices of degree three.



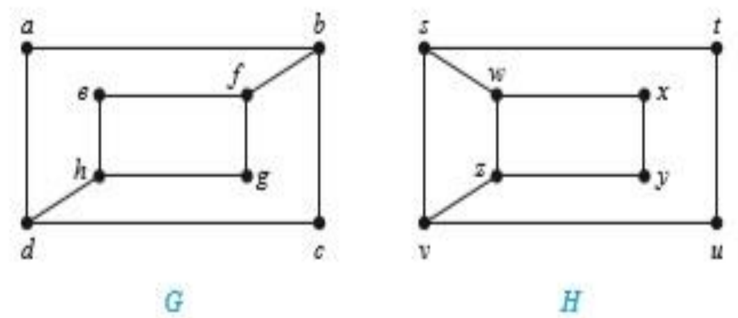
ONE TO ONE CORRESPONDENCE OF VERTEX SET	ONE TO ONE CORRESPONDENCE OF EDGE SET
$u_1 \leftrightarrow v_6$	$(u_1, u_2) \leftrightarrow (v_6, v_3)$
$u_2 \leftrightarrow v_3$	$(u_2, u_3) \leftrightarrow (v_3, v_4)$
$u_3 \leftrightarrow v_4$	$(u_3, u_4) \leftrightarrow (v_4, v_5)$
$u_4 \leftrightarrow v_5$	$(u_4, u_5) \leftrightarrow (v_5, v_1)$
$u_5 \leftrightarrow v_1$	$(u_5, u_6) \leftrightarrow (v_1, v_2)$
$u_6 \leftrightarrow v_2$	$(u_6, u_2) \leftrightarrow (v_2, v_3)$
	$(u_1, u_4) \leftrightarrow (v_6, v_5)$

Since one to one correspondence exist between vertex as well as edge set of the given graph, the graph G and H are isomorphic.

Example: Check whether the given graphs are isomorphic or not. (H.W.)

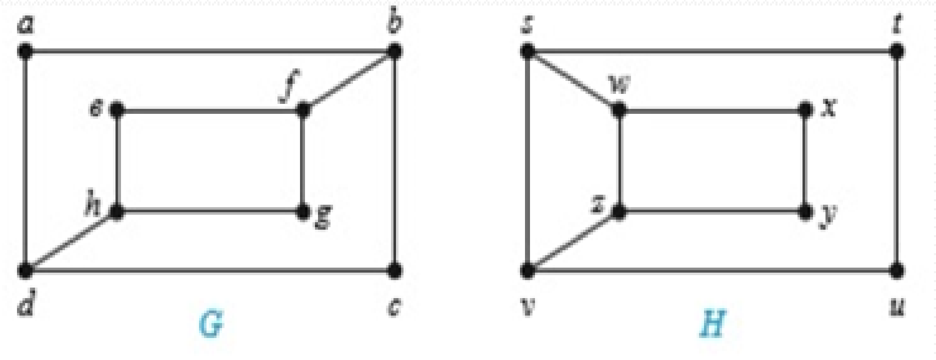


Example: Check whether the given graphs are isomorphic or not.



Solution:

The graphs G and H both have eight vertices and 10 edges. They also both have four vertices of degree two and four of degree three.



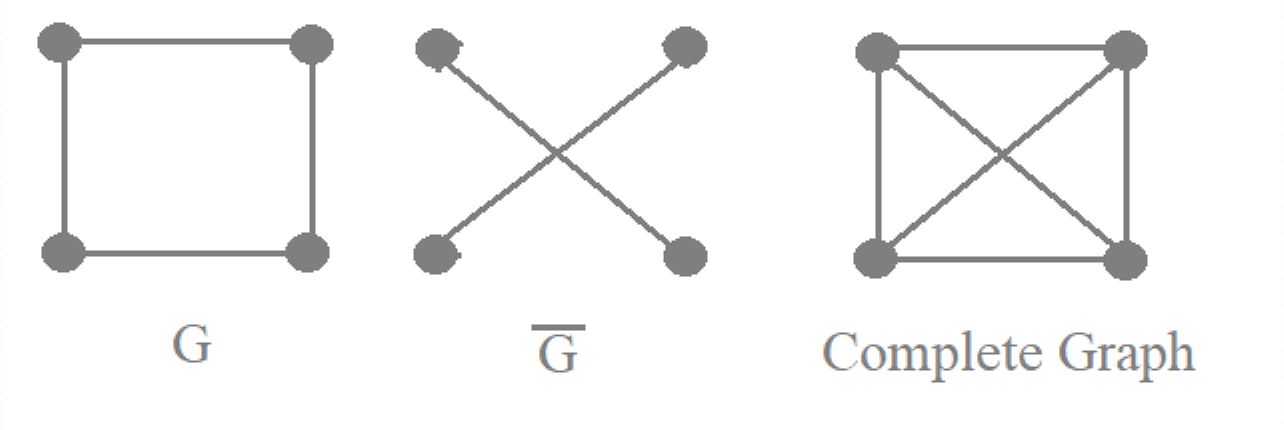
However, G and H are not isomorphic. To see this, note that because deg(a) = 2 in G, a must correspond to either t, u, x, or y in H, because these are the vertices of degree two in H. However, each of these four vertices in H is adjacent to another vertex of degree two in H, which is not true for a in G.
Since One to One correspondence between vertexes does not exit, the graphs G and H are not isomorphic**.**

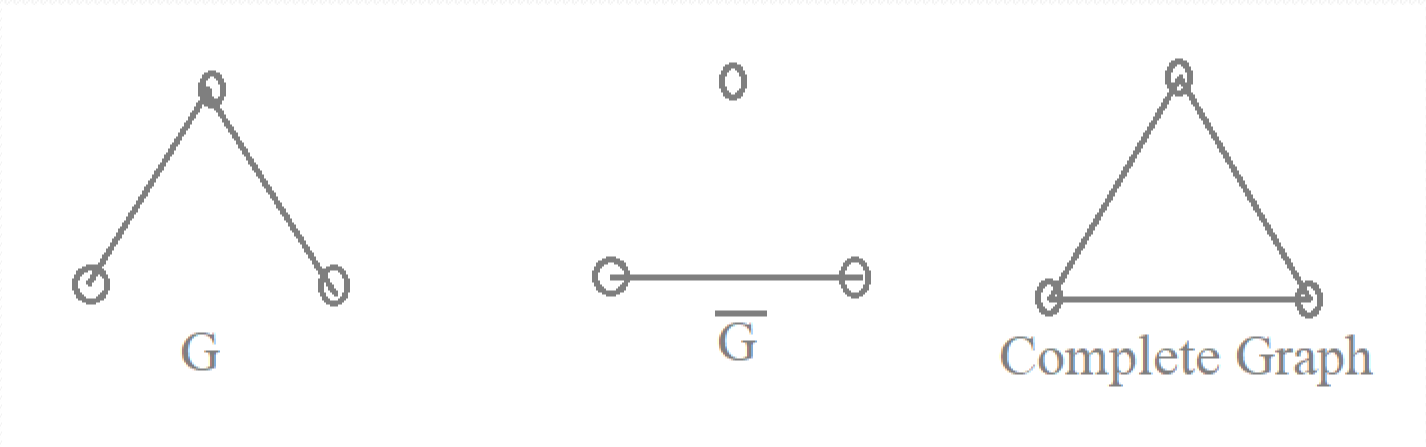
.

COMPLEMENT OF A GRAPH:

If the edges that exist in graph I are absent in another graph II, and if both graph I and graph II with same vertices are combined together to form a complete graph, then graph I and graph II are called complements of each other.

The Complement of Graph G is denoted by $\overline{G}$.





<u>SELF COMPLEMENTARY GRAPH</u>:

A simple graph is said to be self complementary if it is isomorphic to its own complement.

Remarks:

(1) Let G be a simple graph with n vertices and $\overline{G}$ be its complement. Then for each vertex v in G, we have

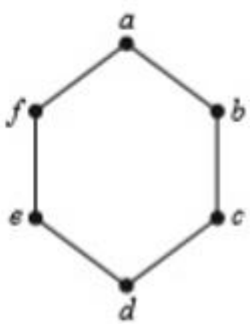
$$d_G(v) + d_{\overline{G}}(v) = n-1, n \text{ is the number of vertices in } G.$$

(2) Let G be a simple graph with n vertices and $\overline{G}$ be its complement. Suppose G has exactly one even vertex then G have (n-1) odd vertices.

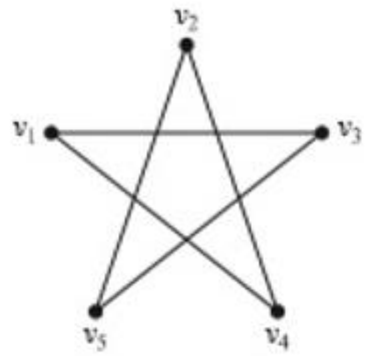
(3) The complement of graph G with n vertices can be obtained from the complete graph K_n by rubbing out all the edges of G.

i.e. Null graph generated from Complete graph G BY REMOVING EDGES is a complement graph of G.

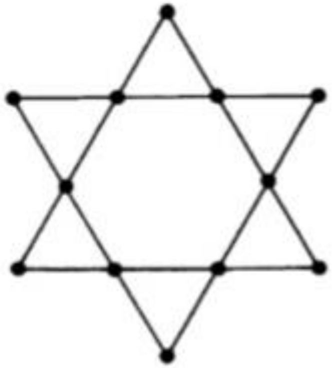
Example: Find the complement of the graph and also find whether it is self complementary or not.



G_1

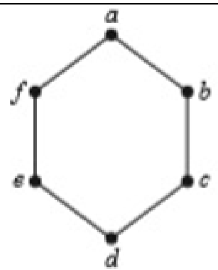
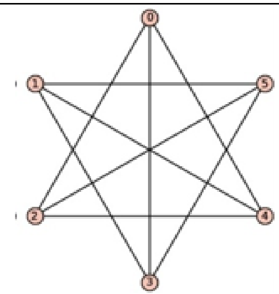
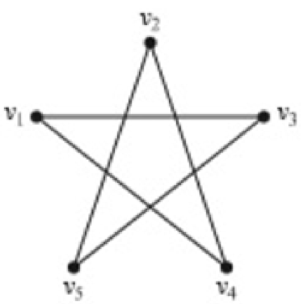
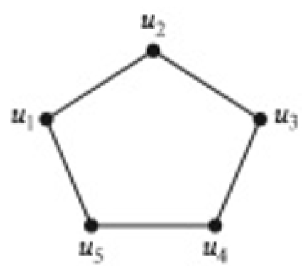
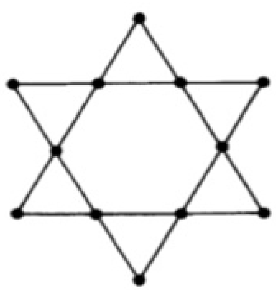
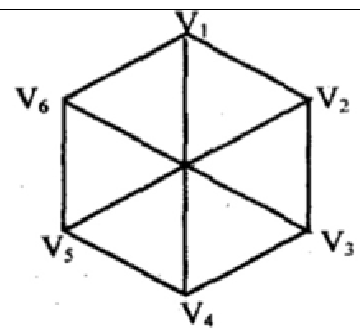


G_2



G_3

Solution:

GIVEN GRAPH	COMPLEMENT OF GIVEN GRAPH	SELF COMPLEMENTARY GRAPH
 G_1		NOT SELF COMPLEMENTARY
		SELF COMPLEMENTARY (In exam you have to show that G and G^1 are isomorphic)
G_2		
 G_3		NOT SELF COMPLEMENTARY

MATRIX REPRESENTATION OF GRAPH

A diagrammatic representation of a graph has limited usefulness; such representation is only possible when the number of vertices and number of edges are reasonably small. Also, their representation in computer memory requires lot of storage space. So a matrix is a convenient and useful way of representing a graph to a computer. There are two different ways of representing a graph inside a computer namely by using

1. Adjacency Matrix
2. Incidence Matrix

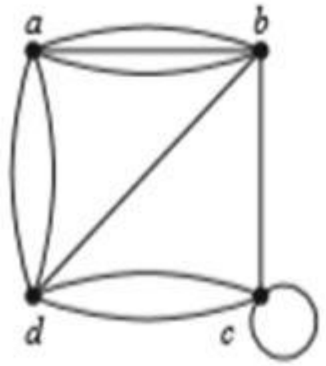
ADJACENCY MATRIX FOR UNDIRECTED GRAPH:

Let G be a graph with n vertices listed as v_1, v_2,..., v_n. The adjacency Matrix A(G) of G with respect to this particular listing of vertices of G is the $n \times n$ matrix $A(G)=[a_{ij}]$ where $(i,j)^{th}$ entry, a_{ij} , is the no. of edges joining the vertex v_i to v_j.

REMARKS:

1. In A(G) we have $a_{ij}=a_{ji}$ for each i and j i.e., adjacency matrix for a graph is symmetric.
2. If G has no loops then all the entry of the main diagonal of A(G) are zero.
3. For a simple graph, the entries of A(G) are either zero or one.

Ex: Write the adjacency matrix for following graph:

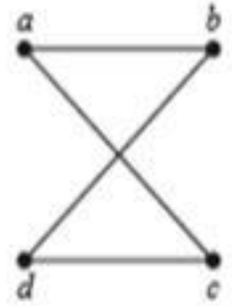


Solution:

The required adjacency matrix is

$$A(G) = \begin{array}{c c} & \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{bmatrix} 0 & 3 & 0 & 2 \\ 3 & 0 & 1 & 1 \\ 0 & 1 & 1 & 2 \\ 2 & 1 & 2 & 0 \end{bmatrix} \end{array}$$

Ex: Write the adjacency matrix for following graph:



Solution:

The required adjacency matrix is

$$A(G) = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

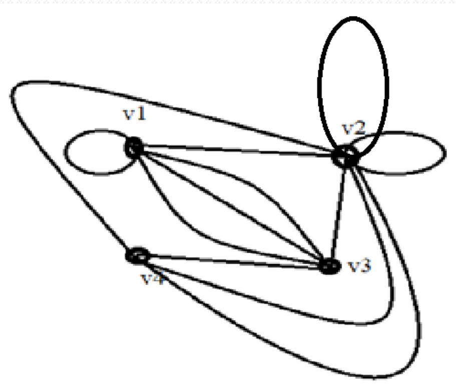
Example: Determine the number of loops and multiple edges in a multi graph G from its adjacency matrix:

$$A(G)=\begin{bmatrix}1 & 1 & 3 & 0\\ 1 & 2 & 1 & 3\\ 3 & 1 & 0 & 1\\ 0 & 3 & 1 & 0\end{bmatrix}$$

Solution:

Since adjacency matrix A is a square matrix of order 4, graph G has four vertices v_1, v_2,$v_{3,}$ v_4. The diagonal of A is indicating the vertices having loops because these entries indicate the number of edges originating and terminating at the same vertex. Thus, there are three loop: one at v_1 and two at v_2 .

Multi edges are 6.



INCIDENCE MATRIX FOR UNDIRECTED GRAPH:

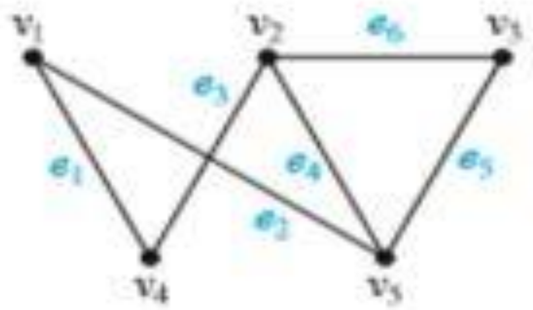
Let v_1, v_2,v_3,...,v_n be the n vertices of a graph G and e_1, e_2, e_3,.....e_m be the m edges of G. Then the incidence matrix of G w.r.t to this particular listing of vertices and edges is $n \times m$ matrix(vertices $\times$ edges). The entries of a matrix can be derived by using

$$I(G) = M(G) = \left[m_{ij} \right] = \begin{cases} 0, & \text{if } v_i \text{ is not end point of } e_j \\ \\ 1, & \text{if } v_i \text{ is end point of } e_j \end{cases}.$$

REMARKS:

1. The sum of the elements in the i^{th} row of *I(G)* gives the degree of vertex v_i.(v_i does not contain loop).
2. The sum of the elements in each column is 2 (corresponding to 2 ends of the edges).

Ex: Write the incidence matrix for following graph:

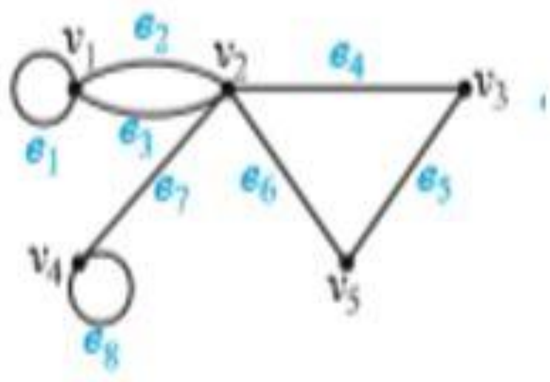


Solution:

The required incidence matrix is

$$I(G)=\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Ex: Write the incidence matrix for following graph:



Solution:

The required incidence matrix is

$$
I(G)=\begin{array}{c}
\\ v1 \\ v2 \\ v3 \\ v4 \\ v5
\end{array}
\begin{array}{c}
\begin{array}{cccccccc} e1 & e2 & e3 & e4 & e5 & e6 & e7 & e8 \end{array} \\
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0
\end{bmatrix}
\end{array}
$$

MATRIX REPRESENTATION OF DIRECTED GRAPH

ADJACENCY MATRIX FOR DIRECTED GRAPH:

For a directed graph G consists of n vertices, an $n \times n$ adjacency matrix A=[a_{ij}] is defined as:

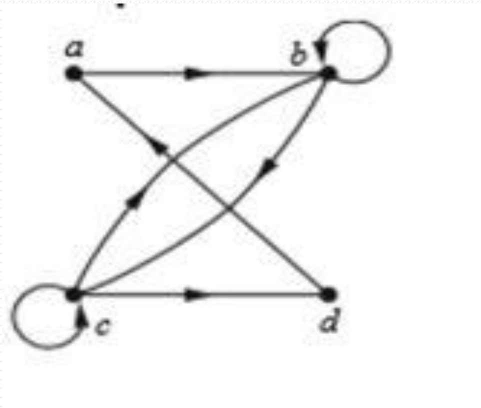
$$a_{ij} = \begin{cases} m & if\ m\, edges\, begining\, at\, vertex\, v_i\ and\ ending\ at\ v_j \\ 0 & ,otherwise \end{cases}$$

INCIDENCE MATRIX FOR DIRECTED GRAPH:

For a directed graph G consists of n vertices and m edges, an $n \times m$ incidence matrix M=[m_{ij}] is defined as :

$$m_{ij} = \begin{cases} 1 & ,if\ v_i\ is\ starting\ vertex\ of\ edg\ e_j \\ -1 & ,if\ v_i\ is\ ending\ vertex\ of\ edge\ e_j \\ 0 & ,if\ v_i\ neither\ starting\ nor\ ending\ vertex\ of\ edge\ e_j \end{cases}.$$

Ex: Write the adjacency and incidence matrix for following graph:

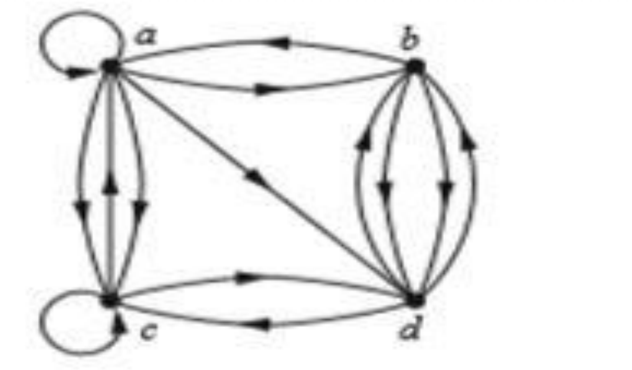


Solution:

The required adjacency and incidence matrix are

$$A(G)=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Ex: Write the adjacency and incidence matrix for following graph:

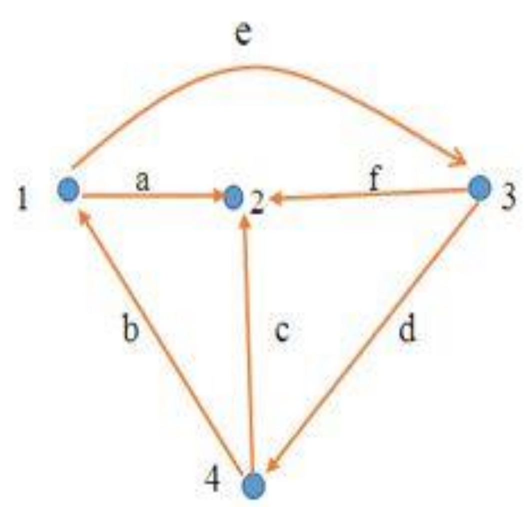


Solution:

The required adjacency and incidence matrix are

$$A(G)=\begin{bmatrix}1 & 1 & 2 & 1\\ 1 & 0 & 0 & 2\\ 1 & 0 & 1 & 1\\ 0 & 2 & 1 & 0\end{bmatrix}$$

Ex: Write the incidence matrix for following graph:



Solution:

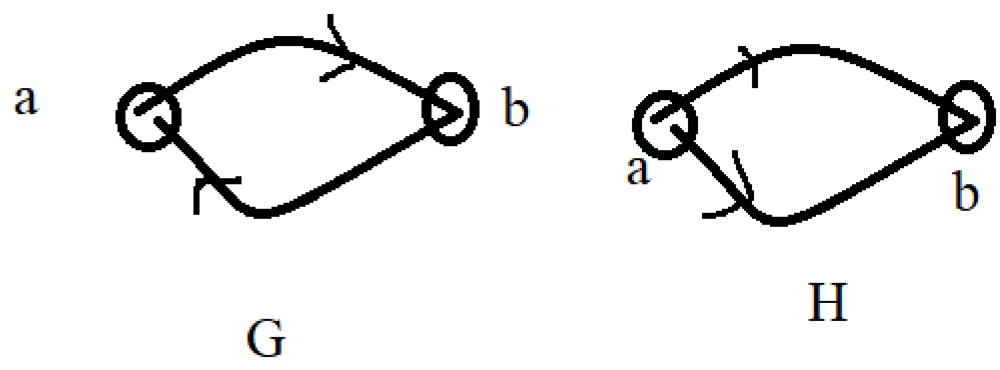
The required incidence matrix is

$$M = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{bmatrix} 1 & -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 & 1 \\ 0 & 1 & 1 & -1 & 0 & 0 \end{bmatrix} \end{array}$$

Remarks:

(1) The Adjacency matrix of an Undirected graph is always Symmetric matrix.

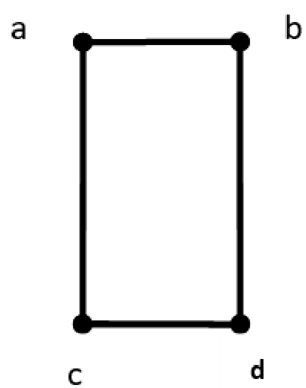
(2) The Adjacency matrix of a directed graph may or may not be Symmetric matrix.



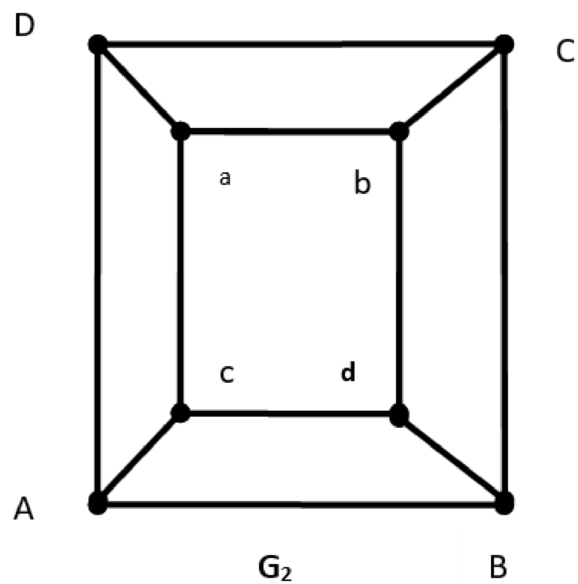
$$A(G) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad A(H) = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$

SUBGRAPHS

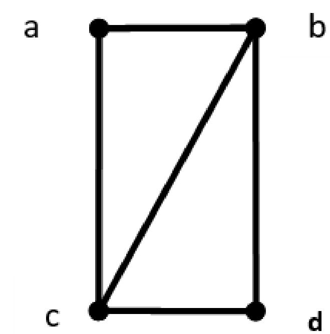
SUBGRAPH: Let H be a graph with vertex set V(H) and edge set E(H) and similarly G be a graph with vertex set V(G) and E(G) .Then H is said to be sub graph of G if $V(H) \subseteq V(G)\,; E(H) \subseteq E(G)$. If H is sub graph of G then it is denoted by $H \subseteq G$ .G is said to be super graph of H.



G_1



G_2



G_3

$G_1 \subseteq G_2$

$G_1 \subseteq G_3$

$G_2 \not\subset G_1$

$G_2 \not\subset G_3$

$G_1 \subseteq G_3$

$G_3 \not\subset G_2$

REMARKS:

(1) Every graph is its own sub graph.

(2) Any simple graph with n vertices is a sub graph of the complete graph K_n.

PROPER SUBGRAPH:

Let H be a sub graph of G i.e., $H \subseteq G$. If $H \subseteq G$, but $H \neq G$ i.e. $V(H) \neq V(G)$ and $E(H) \neq E(G)$ then H is said to be a proper sub graph of G.

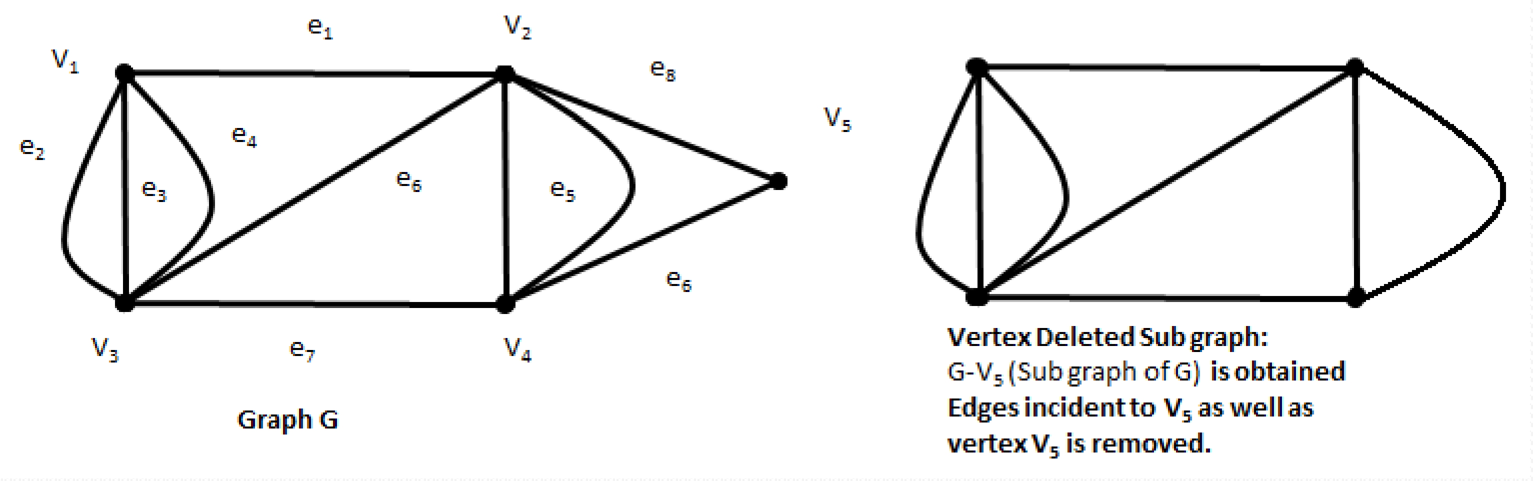
SPANNING SUBGRAPH:

A sub graph H of graph G is said to be a spanning sub graph of G if $V(H) = V(G)$ i.e. sub graph of a graph G is said to be spanning sub graph if it contain all vertices of G.

VERTEX DELETED SUBGRAPH:

Definition 1:

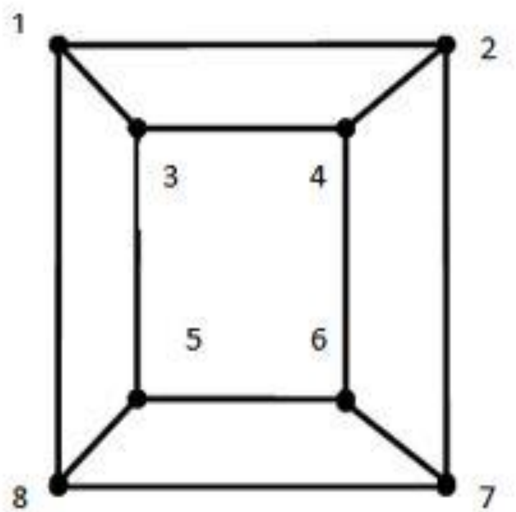
Let G=(V,E) be a graph and V has at least two elements. For any vertex v of G , G-v denotes the sub graph of G with vertex set V-{v} and whose edges are all those edges of G which are not incident with vertex v .i.e., graph G-v is obtained from graph G by removing vertex v and all the edges of G connected to v. Graph G-v is said to be **vertex deleted sub graph.**



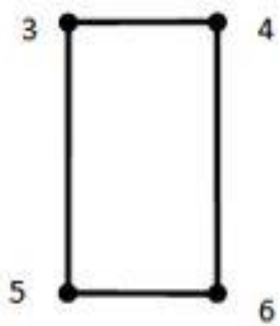
Definition 2:

Let G=(V,E) be the given graph and U be a proper subset of V, then G-U denotes the sub graph of G with vertex set V- U and whose edges are all those of G which are not incident with any vertex of U.

For example,



Graph G



Vertex Deleted Sub graph :

G-V where V={1,2,8,7} (Sub graph of G) **is obtained by removing edges incident to vertex 1,2,8,7 as well as vertices 1,2,8,7.**

<u>EDGE DELETED SUBGRAPH:</u>

Definition 1:

Let G=(V,E) be the given graph and e be the edge of G then G-e denotes the subgraph of G having V as its vertex set and E-{e} as it edge set. i.e. G-e is obtained from G by removing edge e **<u>(but not the end point of e)</u>** G-e is said to be **edge deleted sub graph** of G.

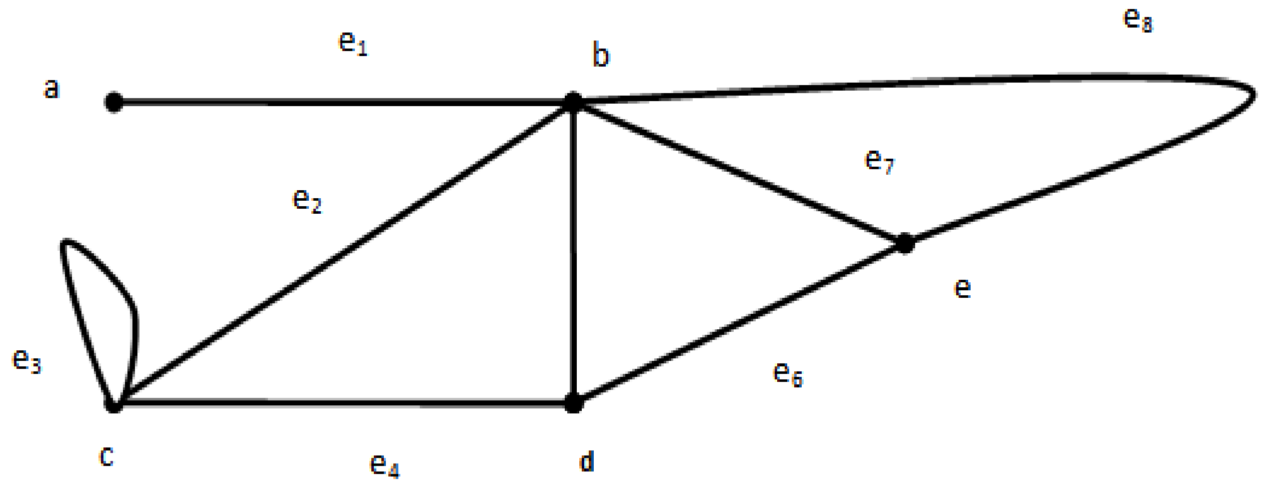


Figure: Graph G

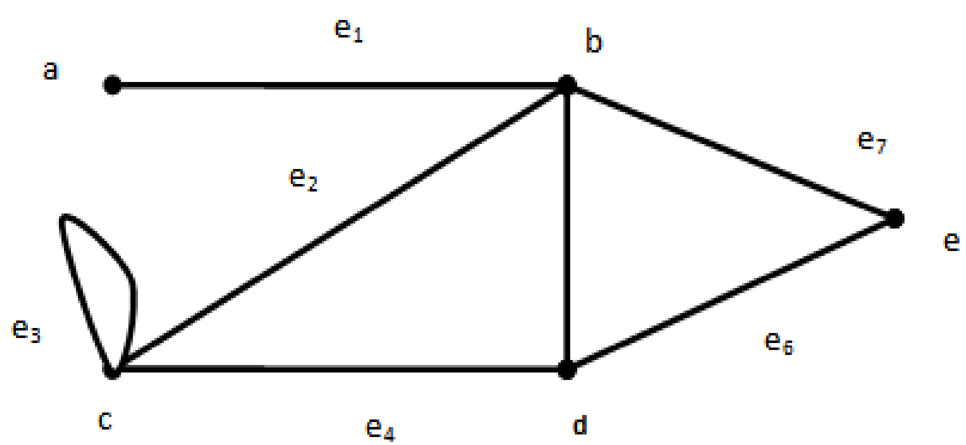


Figure: Edge Deleted Sub graph: G-e_8 is obtained from graph G **by removing only edge and not end points of the edge.**

Definition 2:

Let G =(V,E) be the given graph. F be a subset of edge set E. G-F denoted the sub graph of G with vertex set V and edge set E-F i.e. G-F is obtained by deleting all the edges in F **but not their end point.** G-F is also said to be **edge deleted sub graph** of G.

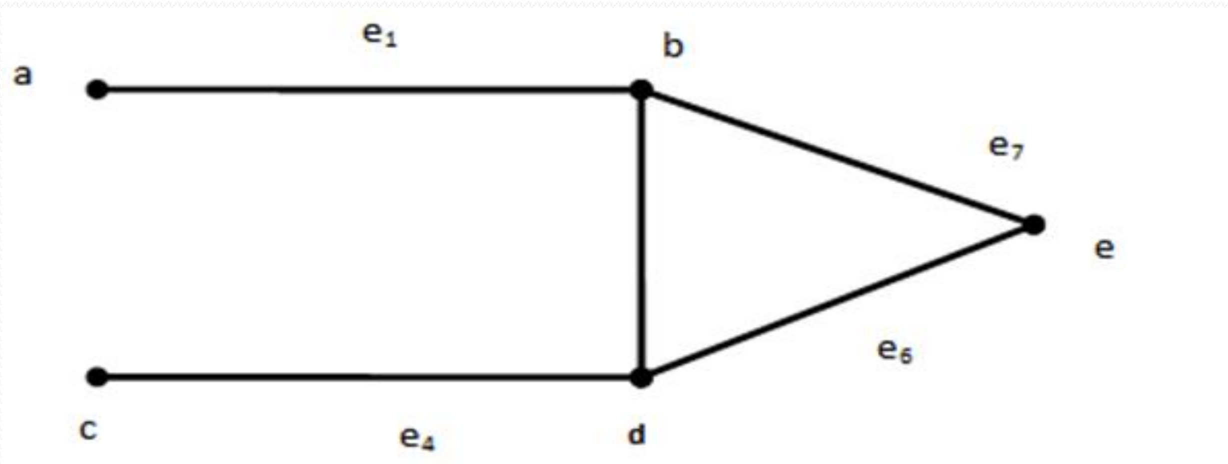


Figure: Edge Deleted Sub graph: G-E where E={e_3,e_2,e_8}is obtained from graph G **by removing only edge and not end points of the edge.**

<u>WALK</u>:

A walk in a graph G is a finite alternating sequence of vertices and edges beginning and ending with vertices such that each edge is incident with vertices preceding and following it.

Example: $W = v_0 e_1 v_1 e_2 v_2 e_3 v_3 .... v_{k-1} e_k v_k$

NOTES:

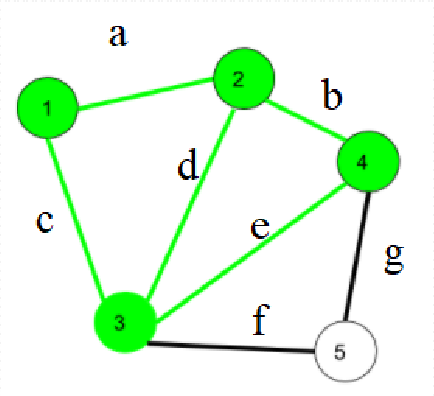
1. The above walk is said to be v_0-v_k walk or walk from v_0 to v_k.
2. The vertex v_0(Starting vertex of walk) is said to be origin of the walk W and vertex v_k (End vertex of walk) is said to be end of the walk.
3. For a walk W v_0 and v_k need not be distinct i.e., origin and end need not be distinct.
4. Vertices $v_1, v_2, v_3 ...., v_{k-1}$ are said to be internal vertices for the walk W.
5. **The number of edges in the walk W is said to be length of walk.**
6. In a simple graph a walk $W = v_0 e_1 v_1 e_2 v_2 e_3 v_3 .... v_{k-1} e_k v_k$ is determined by sequence $W = v_0 v_1 v_2 v_3 .... v_{k-1} v_k$ **of** vertices because for each pair $v_i v_{i-1}$ there is only one possible edge.
7. **In a walk there may be repetition of vertices and edges.**

Open walk:

A walk is said to be an open walk if the starting and ending vertices are different i.e. the origin vertex and terminal vertex are different.

Closed walk:

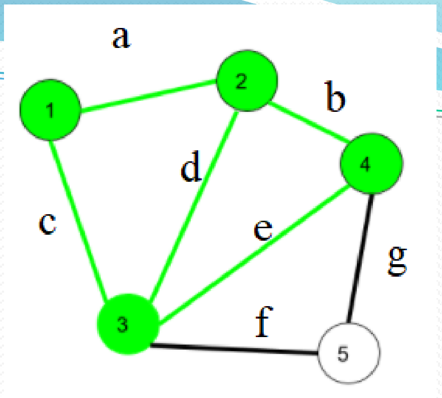
A walk is said to be a closed walk if the starting and ending vertices are identical i.e. if a walk starts and ends at the same vertex, then it is said to be a closed walk.



In the above diagram:

W=1a2d3e4g5f3 is an open walk. Length of a W is 5.

W=1a2d3e4g5f3c1 is a closed walk. Length of a W is 6.



TRIVIAL WALK:

A walk containing no edge is said to be trivial walk. For an vertex v, W=v is said to be trivial walk . Length of trivial walk is zero.

In above example, W=1 is a trivial walk.

TRAIL:

A Trail is a walk in which no edges are repeated.

In the above example, the walk W=1a2d3e4g5 is a trail.

REMARK:

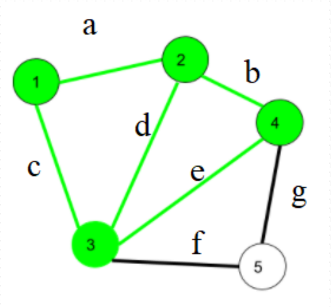
Every trail is a walk but every walk is not a trail.

For example, the walk 1a2d3c1a2 is a walk but not trail.

PATH:

A path is a walk in which no vertex is repeated.

The walk 1a2d3e4g5 is a path.



REMARKS:

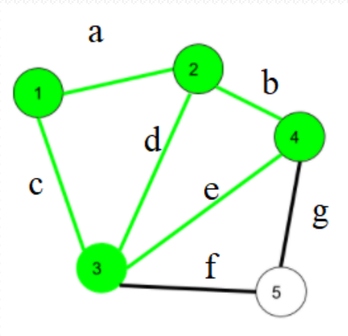
- It is clear that in path no vertex is repeated.
- A path with n vertices is denoted by P_n and has length is n-1.
- Every path is a walk but each walk is not a path.
- A loop can be included in walk but not in a path.

Cycle or Circuit:

In Graph theory, a **cycle in a graph** is a non-empty trail in which the only repeated vertices are the first and last vertices.

K-CYCLE:

A cycle of length k is said to be K- cycle i.e. A cycle with K edge are said to be K cycle.



e.g. W=3c1a2d3 is a cycle and its length is 3.

e.g. C_3 =3c1a2d3 is a 3-cycle

Remarks:

- If K is odd then K- cycle is said to be an odd and K is even then K-cycle is said to be even.
- An n-cycle is denoted by C_n.
- A loop is just a 1-cycle.

<u>CONNECTED VERTEX:</u>

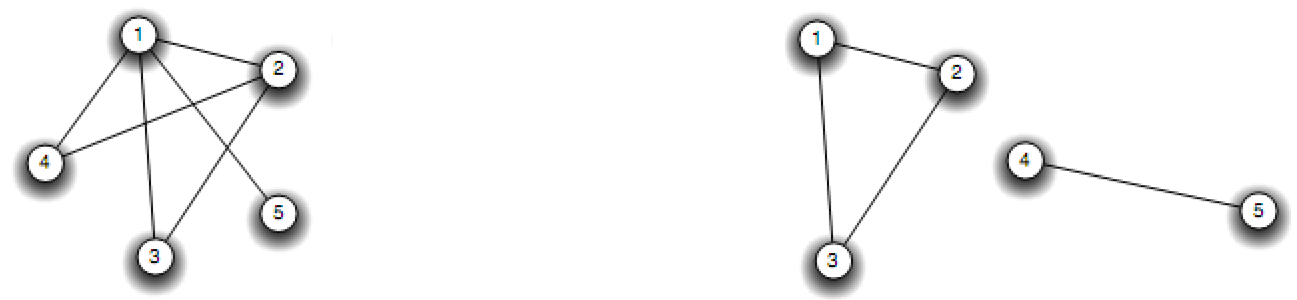
A vertex u is said to be connected to vertex v in a graph G if there is a path in G from u to v.

<u>NOTES:</u>

1. An vertex u is connected to itself by trivial path P=u.
2. If u is connected to v and v is connected to w then u is connected to w.

Connected undirected graph and Disconnected undirected graph:

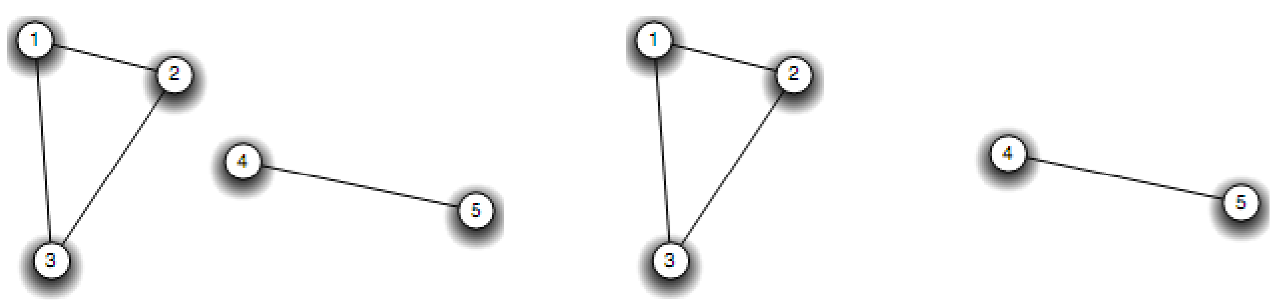
A graph G is said to be connected if there is at least one path between every pair of vertices of G otherwise graph G is said to be disconnected.



(a) Connected (b) Disconnected

Note:

Every disconnected graph can be split into number of connected subgraphs, called connected components.



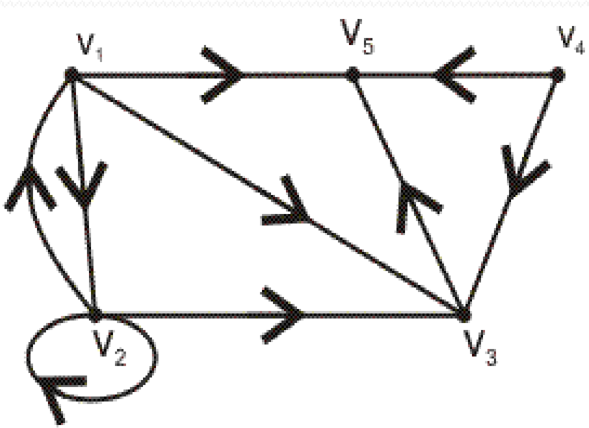
Disconnected Connected + connected

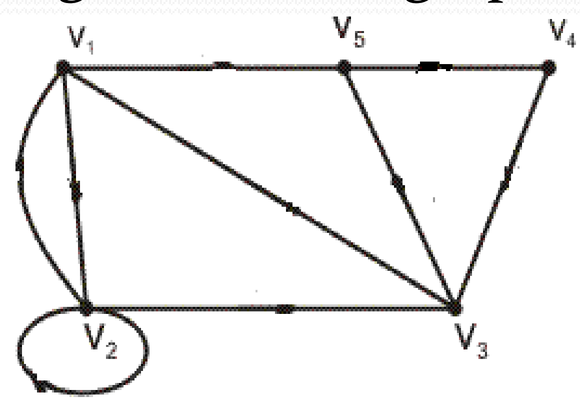
CONNECTEDNESS FOR DIRECTED GRAPH

We cannot use the same criteria to determine the connectivity of a directed graph as that for an undirected graph.

PATH: A path is a walk in which no vertex is repeated.

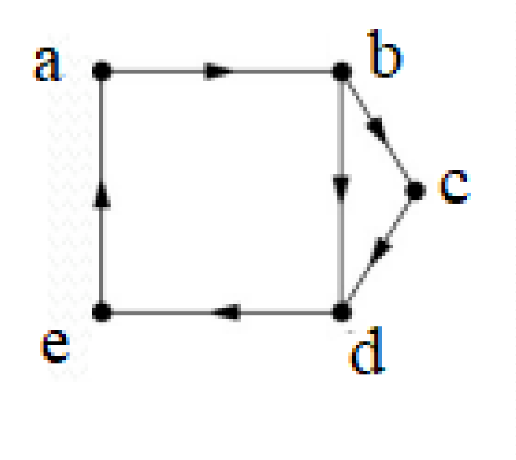
WEAKLY CONNECTED GRAPH:
A directed graph is weakly connected if there is a path between every two vertices in the corresponding undirected graph.





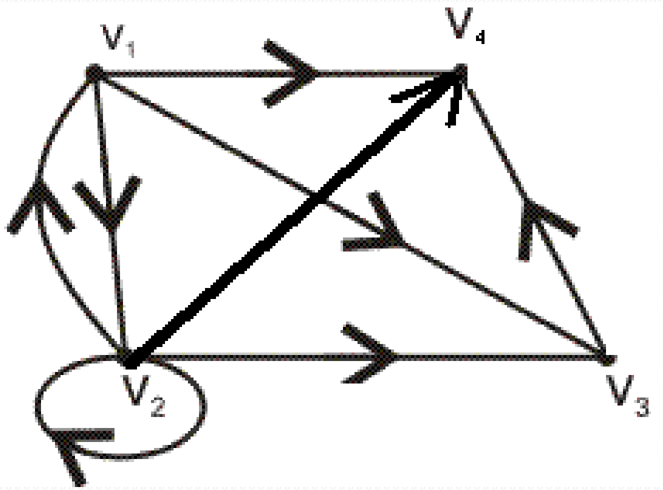
STRONGLY CONNECTED:

A directed graph is called strongly connected if for any pair of vertices u and v, there exist a path from u to v as well as v to u.



UNILATERALLY CONNECTED:

A directed graph is called unilaterally connected if given any two vertices of the graph there exist a path from one vertex to another although a reverse path does not necessarily exist.



(v1 to v3 path exists but v3 to v1 path does not exist)

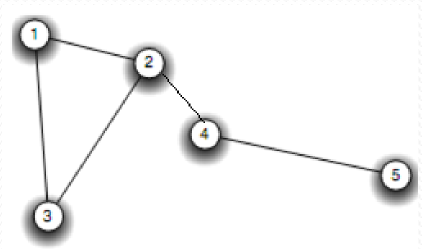
REMARKS:

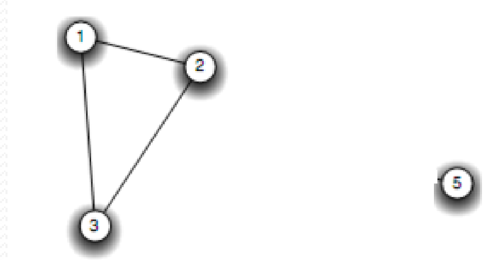
1. Every strongly connected graph is unilaterally connected but converse is not true.
2. Every strongly connected graph is weakly connected but converse is not true.

VERTEX CONNECTIVITY AND EDGE CONNECTIVITY

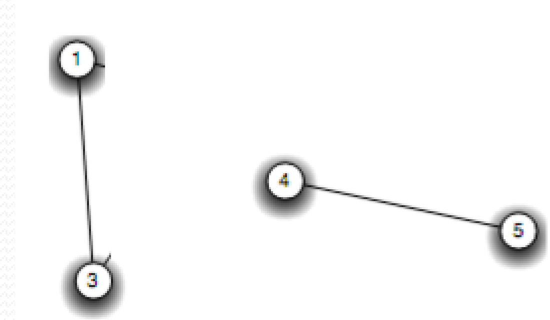
CUT VERTEX OR ARTICULATION POINT:

Let G be a connected graph. Then a cut vertex v is a vertex whose removal from G results in a disconnected graph.





Cutvertex is 4 or cutvertex is 2



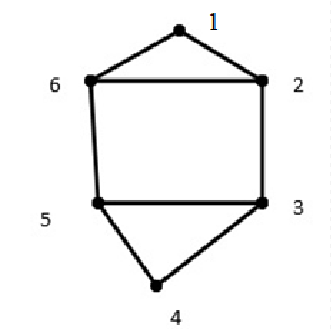
Remarks:

1. Not all graph have cut vertices.
2. The complete graph K_n , $n \geq 3$ has no cut vertices.

VERTEX CONNECTIVITY:

Let G be a connected graph. The minimum number of vertices in G whose deletion from G leaves either a disconnected graph or complete graph with one vertex is said to be vertex connectivity of G.

It is denoted by $\kappa(G)$.



The graph G given in the above figure has no cut vertex. But a disconnected sub graph is obtained when two of these vertices 2 and 6 or 2 and 5 are deleted.

Therefore for this graph G , $\kappa(G)$(vertex connectivity) is 2.

REMARKS:

1.For $n \geq 2$ the deletion of any vertex from K_n result in K_{n-1} and in general the deletion of t vertices (t<n) result in K_{n-t} therefore , $\kappa(G)$=n-1.

2. A connected graph G has $\kappa(G)$=1 iff G=K_2 (i.e. complete graph with 2 vertices) or G has a cut vertex.

3. $\kappa(G)$= 0 iff G=K_1 or G is disconnected.

CUT EDGE(BRIDGE) OR ISTHMUS:

A bridge is an edge whose removal increases number of connected components of given graph. i.e., removal of an edge in a graph results in two or more graphs.

REMARK: It is not necessary that all the graph have cut edge.

EDGE CONNECTIVITY:

Let G be a connected graph. The minimum number of edges in G whose deletion from G leaves either a disconnected graph or complete graph with one vertex is said to be edge connectivity of G.

It is denoted by $\lambda(G)$.

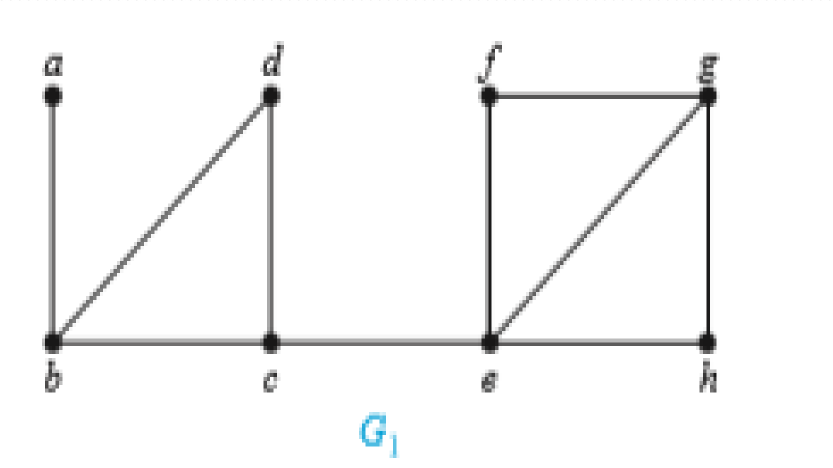
REMARKS:

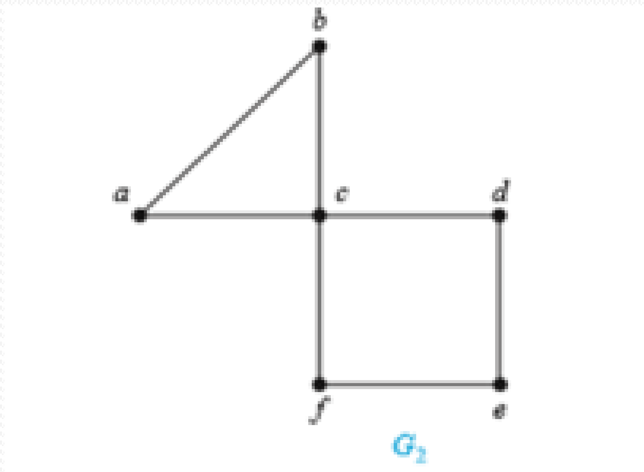
(1) G is disconnected graph iff $\lambda(G) = 0$.

(2) G is disconnected graph, $\kappa(G)$= 0.

(3) For any graph G, $\kappa(G) \leq \lambda(G)$.

Example : Find the cut vertices , cut edges, vertex connectivity and edge connectivity in the following graph:





Solution:

For G_1,

Cut Vertices: b, c, e

Cut Edges: (c,e)

Vertex connectivity: 1

Edge connectivity: 1

PLANAR GRAPHS

A graph is said to be **planar** if there exists some geometric representation of G which can be drawn on a plane such that no two of its edges crossovers to each other.

OR

A graph that can be drawn on a plane without a crossover between it edges is called planar graph.

REGION OF A GRAPH:

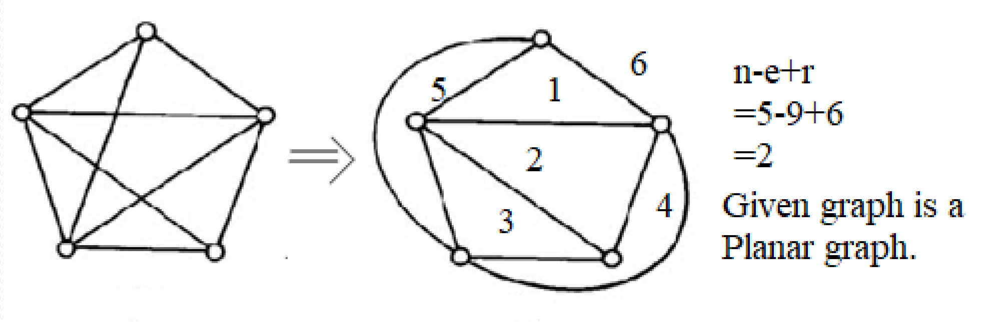
A region of a planar graph is defined to be an area of the plane that is bounded by edges and is not further divided into subareas.

Note: we include the outside area of a graph as region.

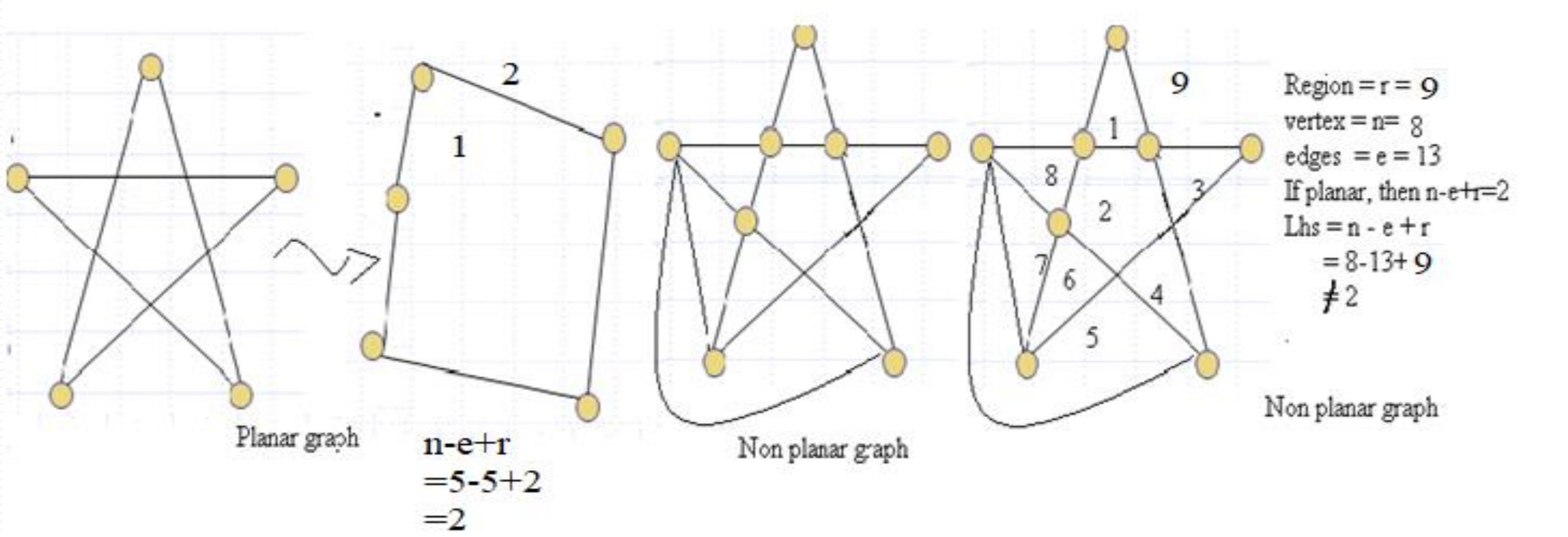
EULER'S FORMULA:

If a connected planar graph G has 'n' vertices, 'e' edges and 'r' number of regions, then $n - e + r = 2$

Examples,



5
1
6
2
3
4
n-e+r
=5-9+6
=2
Given graph is a
Planar graph.



Planar graph
2
1
n-e+r
=5-5+2
=2
Non planar graph
9
1
8
2
3
7
6
4
5
Region = r = 9
vertex = n= 8
edges = e = 13
If planar, then n-e+r=2
Lhs = n - e + r
= 8-13+ 9
≠ 2
Non planar graph

Result-1:

If a planar graph has ‘k’ number of connected components, then $n - e + r = k + 1$.

Result-2:

If G is connected simple planar graph with

$n \geq 3$ vertices and ‘e’ edges, then $e \leq 3n - 6$

Result-3:

If G is connected simple planar graph with n vertices and ‘e’ edges and $e \leq 2n - 4$

Example: A connected planar graph has 10 vertices each of degree 3. Into how many regions does a representation of this planar graph split the plane?

Solution:

Here $n = 10$ and degree of each vertex is 3.

Hence using first theorem of graph theory, the sum of degrees of all the vertices will be 30, so number of edges in a graph is 15.

Using Euler's formula,

$$n - e + r = 2$$

$$\therefore 10 - 15 + r = 2$$

$$\therefore r = 7$$

Cycle or Circuit : In **graph** theory, a **cycle in a graph** is a non-empty trail in which the only repeated vertices are the first and last vertices.

TRAIL: A Trail is a walk in which no edges are repeated.

EULERIAN AND HAMILTONIAN GRAPH

EULERIAN PATH:

An **Eulerian trail** (or **Eulerian path**) is a trail in a finite graph that visits every edge exactly once (Repetition of vertices is allowed. Path in title is a word and it is not a definition of a Path)

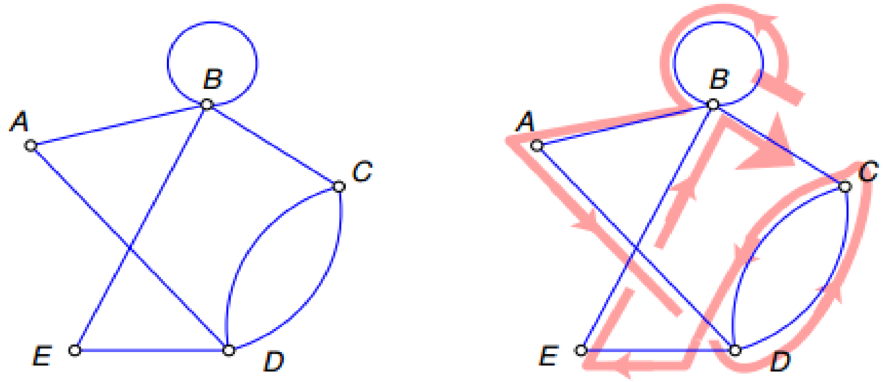
EULERIAN CIRCUIT:

Eulerian circuit or **Eulerian cycle** is an Eulerian trail that starts and ends on the same vertex.

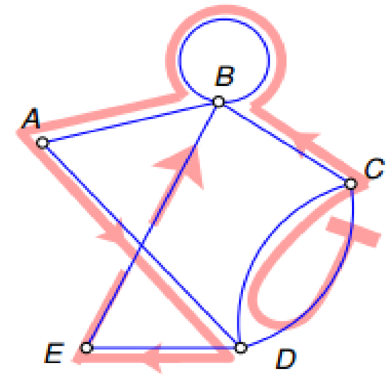
EULERIAN GRAPH:

Eulerian graph is a graph which contains at least one Eulerian circuit.

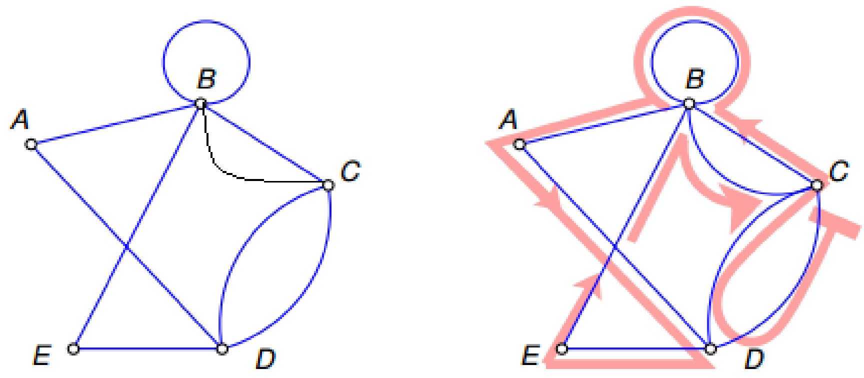
FOR EXAMPLES,



An Euler path: BBADCDEBC



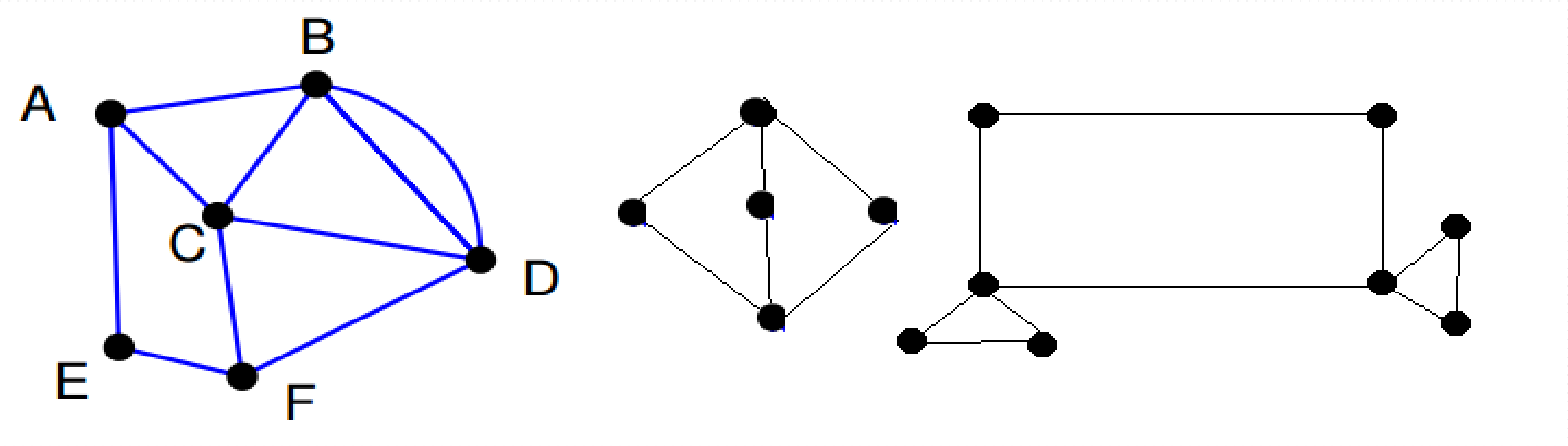
Euler path: CDCBBADEB



An Euler circuit: CDCBBADEBC

EXERCISE

EXAMPLE: Check that Euler circuits and paths exit or not if yes then find.



THEOREM: An undirected graph is an Eulerian graph if and only if all the vertices are even.

THEOREM: A directed graph is an Eulerian graph if and only if in degree and out degree of for all the vertices are same.

HAMILTONIAN PATH:

Hamiltonian path in a graph G is a path which passes through all the vertices of a graph exactly once.(Repetition of edges is not allowed)

HAMILTONIAN CIRCUIT:

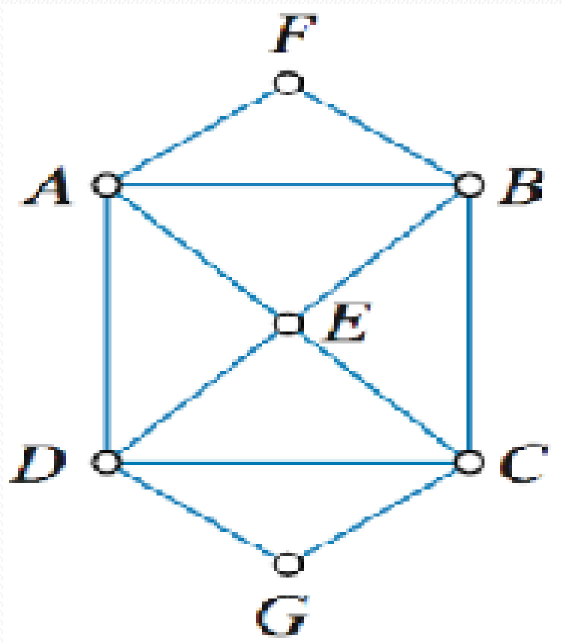
Hamiltonian circuit in a graph G is a circuit which passes through all the vertices of a graph exactly once. Also, initial and terminal vertex must be same. (Repetition of edges is not allowed)

HAMILTONIAN GRAPH:

Hamiltonian graph is a graph which contains at least one Hamiltonian circuit.

EXAMPLE:

The following graph contains Hamiltonian path and circuit.

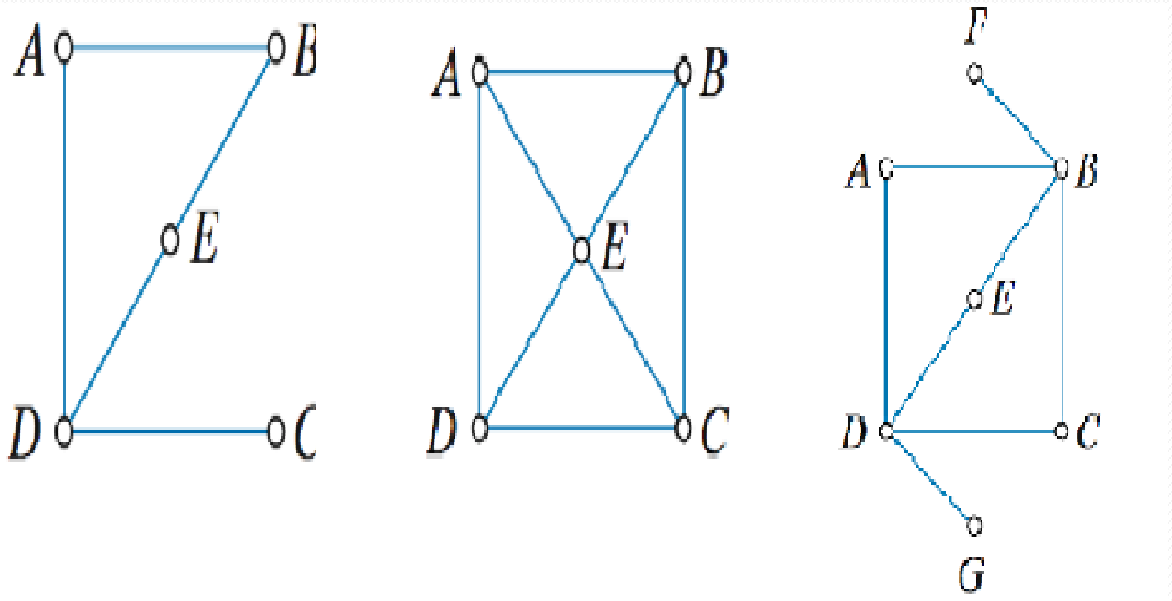


Path : - A F B C G D E ;

Circuit :- A F B C G D E A

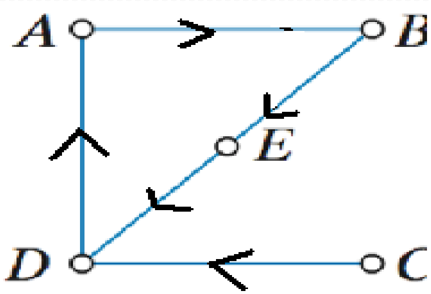
EXERCISE

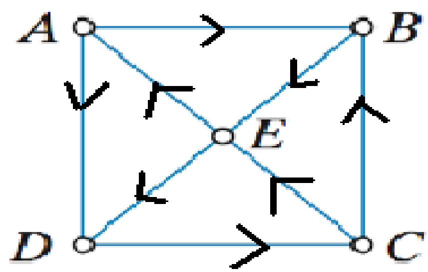
EXAMPLE: Check that Hamiltonian circuits and paths exit or not if yes then find.

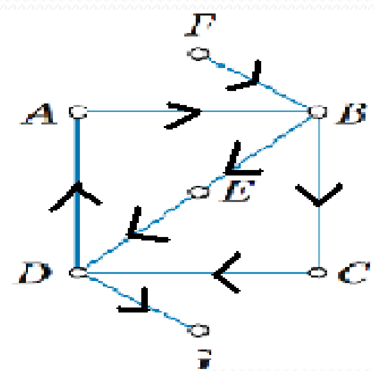


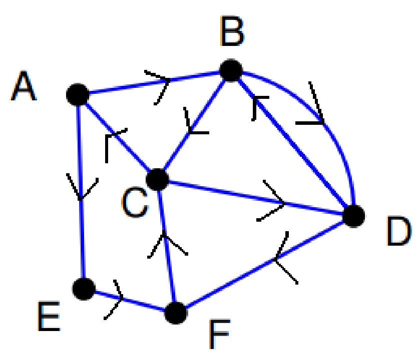
EXERCISE

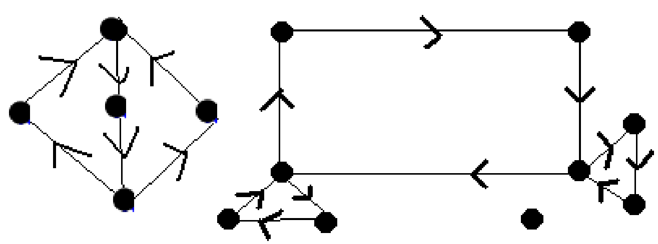
EXAMPLE: Check that Euler path, Euler circuit, Hamiltonian path and Hamiltonian circuit exist or not.









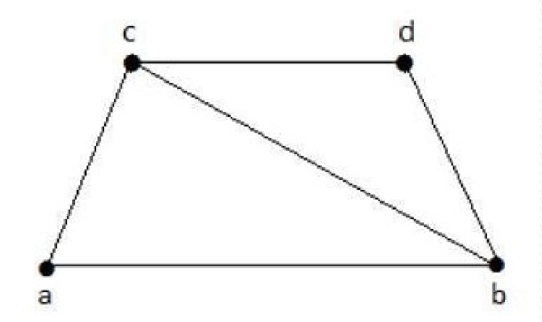


MATCHING IN A GRAPH

Matching Graph:

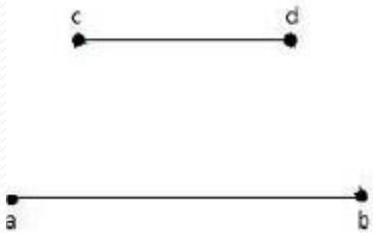
Let G be a graph. A Matching graph M(G) or M is a subgraph of a graph G where there are no edges adjacent to each other. Simply, there should not be any common vertex between any two edges. i.e. each vertex of G is incident with at most one edge in M. Therefore, the vertices of M(G) should have a degree of 1 or 0.

Example: Find the Matching graph of following graph.

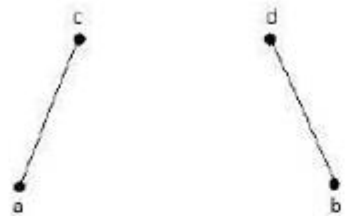


G

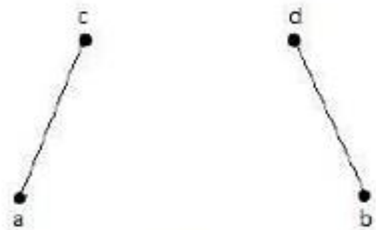
Solution:



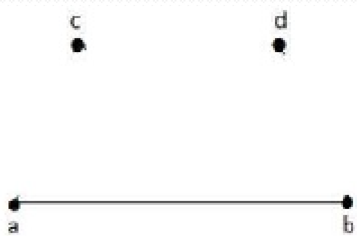
M_1



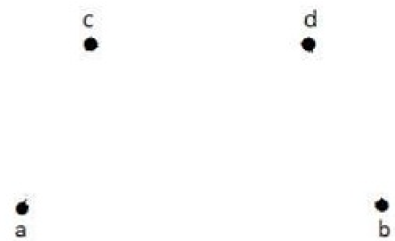
M_2



M_3



M_4



M_5

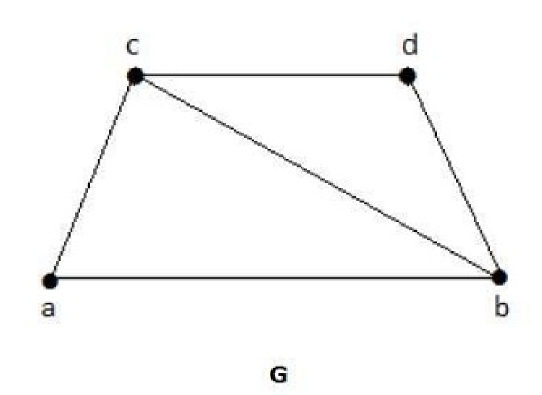
Remarks:

1) Null graph as a sub graph of G is always Matching graph of G.
2) In a matching, if deg(V) = 1, then (V) is said to be matched. if deg(V) = 0, then (V) is not matched.
3) In a matching, no two edges are adjacent. It is because if any two edges are adjacent, then the degree of the vertex which is joining those two edges will have a degree of 2 which violates the matching rule.

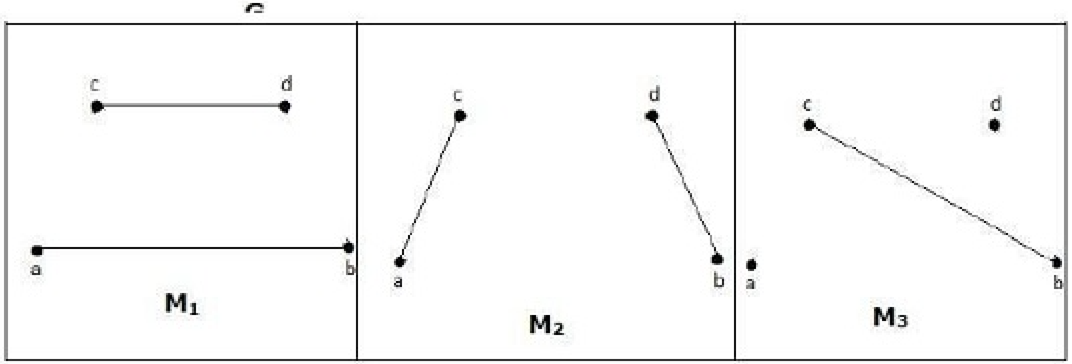
Maximal Matching:

A matching M of graph ‘G’ is said to maximal **if no other edges of ‘G’ can be added to M**.

Example: Find the Maximal Matching graph of following graph.



Solution:



M1, M2, M3 from the above graph are the maximal matching of G.

Maximum Matching:

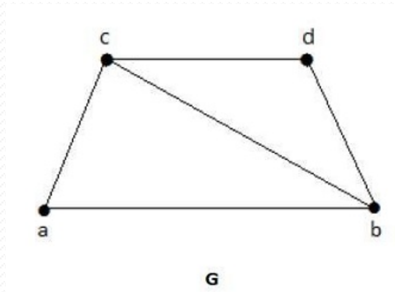
Maximum matching is defined as the maximal matching with maximum number of edges. It is also known as largest maximal matching.

The number of edges in the maximum matching of 'G' is called its **matching number**.

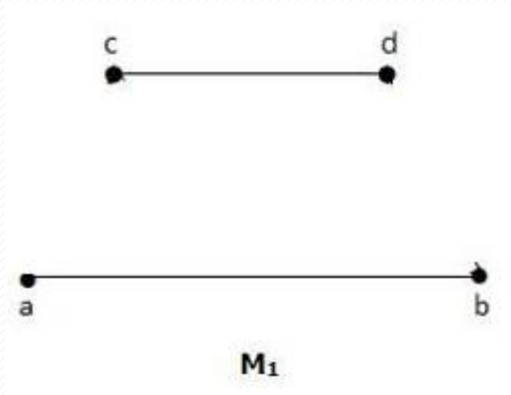
Perfect Matching:

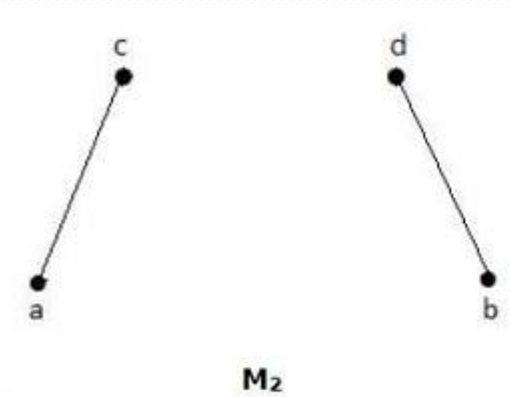
A matching (M) of graph (G) is said to be a perfect match, if $deg(V) = 1$, for every vertex of M.

Example: Find the matching number of following graph.



Solution:





For a graph given in the above example, M1 and M2 are the maximum matching of 'G' and its matching number is 2.

Hence by using the graph G, we can form only the subgraphs with only 2 edges maximum.

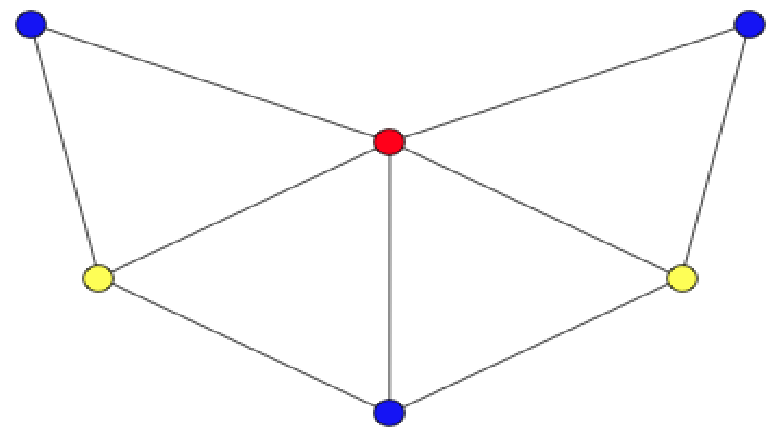
Hence we have the matching number as two. Since degree of each vertex is 1. Therefore it is a perfect matching.

Graph coloring

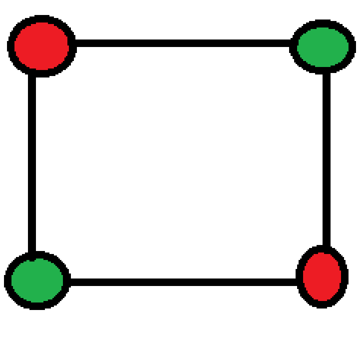
A **graph coloring** is an assignment of labels or colors, to the vertices of a graph such that no two adjacent vertices share the same color.

The **chromatic number** $\chi(G)$ of a graph G is the minimal number of colors for which such an assignment is possible.

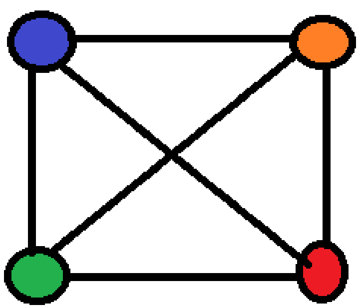
Other types of colorings on graphs also exist, most notably **edge colorings** that may be subject to various constraints.



A graph coloring for a graph with 6 vertices. It is impossible to color the graph with 2 colors, so the graph has chromatic number 3.



The Chromatic number is 2



The Chromatic number is 4.

Remarks:

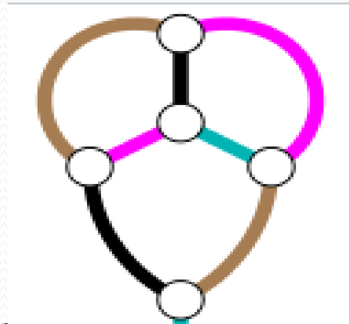
1) The chromatic number of Kn is n because In the complete graph, each vertex is adjacent to remaining (n – 1) vertices. Hence, each vertex requires a new color.

2) The chromatic number of trivial graph is 1.

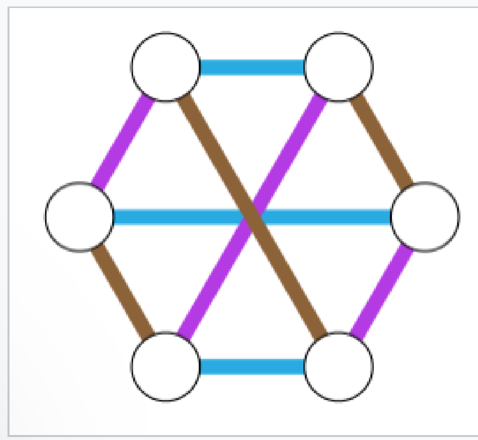
3) The chromatic number of Null graph with n vertices is 1.

Edge coloring:

The most common type of edge coloring is similar to vertex colorings. Each edge of a graph has a color assigned to it in such a way that no two adjacent edges are of the same color. Such a coloring is a **proper edge coloring or edge coloring**. The minimum required number of colors for the edge coloring of a given graph is called the **chromatic index** of the graph.



The **chromatic index is 4.**



The complete bipartite graph $K_{3,3}$ with each of its color classes drawn as parallel line segments on distinct lines.

The **chromatic index is 3.**

APPLICATION OF GRAPH THEORY IN DIFFERENT FIELDS

The ideas and concepts of Graph theory are widely used in various branches of science. In general, without knowing the concepts of graph we also use these in our day to day life. For example when we have to go to a place which is connecting with our starting point by different ways then we use the shortest road to arrive the destination soon. Here if we observe this problem from the point of view of graph theory the two places can be considered as vertices and roads are as edges. If we also consider the direction of travel, then the graph must be directed. Similarly, we can use these concepts of graph theory in various situations. A graph can be used to present almost any physical situation involving discrete and relationship among them. Here the applications of graph theory in various branches of science are mention below.

Graph	Node or Vertices	Edges or Arcs
Communication	Telephone, Computer	Fibber Optic Cable
Internet	Class C Network	Connection
Circuit	Gate, Register, Processor	Wire
Transportation	Street Intersection, Airport	Highway, Airway Route
Social Relation	Person	Friendship,

APPLICATIONS IN GOOGLE MAP

Now a days, Google map is a very useful tool for travelling anywhere in the world. Using Google map we can find all routes from any place to any other place and also can find the shortest route. In case of Google map, we can consider the places as vertices of graph and the routes as the edges. Then the software of google map, when find the routes between two places it find all edges between these two places or vertices and also gives the shortest edge as the shortest path.

APPLICATION IN INTERNET

Internet is a very useful invention of modern science. In the working technique of internet the concepts of graph theory are used. In case of connectivity of internet, all the users are considered as vertices and the connection between them are edges. Then all internet users form a very complicated graph and data and information from one user to another user are shared through the shortest route in between them. Similarly, in case of social networking sites one friend is connected to all of his friend and his friends are also connected to others. If we consider the friends as vertices of graph and define an edge in between them if they are friend then it will be a graph. While using Google to search for Webpages, Pages are linked to each other by hyperlinks. Each page is a vertex and the link between two pages is an edge.

APPLICATION IN COMPUTER SCIENCE

There is a major role of graph theory in computer science. Graph theory concepts are used to develop the algorithm of different programs. Using these algorithms and programmes we can solve different theoretical problems. There are some algorithms listed below.

(1). Shortest path algorithm in a network.

(2). Finding minimum spanning tree.

(3). Finding graph planarity.

(4). Algorithms to find adjacency matrices.

(5). Algorithms to find the connectedness.

(6). Algorithms to find the cycles in a graph etc.

There are many computer languages which helps to solve different problems using graph theory concepts.

Some computer languages available are listed as follows:

(1). GTPL - Graph Theoretic Language

(2). GASP - Graph Algorithm Software Package.

(3). HINT - Extension of LISP.

(4). GRASPE - Another extension of LISP.

(5). DIP - Directed Graph Processor.

(6). An Interactive Graph Theory System - Extension of FORTRAN.

(7). GEA - Graphic Extended ALGOL.

(8). GIRL - Graph Information Retrieval Language.

(9). FGRAAL - FORTRAN Extended Graph Algorithmic Language.

TO CLEAR ROAD BLOCKAGE:

When roads of a city are blocked due to ice. Planning is needed to put salt on the roads. Then Euler paths or circuits are used to traverse the streets in the most efficient way.